US005880731A

United States Patent [19]

Liles et al.

[11] **Patent Number: 5,880,731**

[45] **Date of Patent: *Mar. 9, 1999**

[54] **USE OF AVATARS WITH AUTOMATIC GESTURING AND BOUNDED INTERACTION IN ON-LINE CHAT SESSION**

[75] Inventors: **Christopher A. Liles**, Seattle; **Manuel Vellon**, Bellevue, both of Wash.

[73] Assignee: **Microsoft Corporation**, Redmond, Wash.

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **572,307**

[22] Filed: **Dec. 14, 1995**

[51] **Int. Cl.^{6}** **G06F 3/00**

[52] **U.S. Cl.** **345/349**; 345/358; 345/473; 345/330

[58] **Field of Search** 395/329, 330, 395/331, 332, 348, 349, 358, 806, 807, 957, 960, 972; 345/329, 330, 331, 332, 348, 349, 358, 302, 957, 960, 972, 473

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,257,306	10/1993	Watanabe	395/330 X
5,347,306	9/1994	Nitta	395/330 X
5,491,743	2/1996	Shiio et al.	395/330 X
5,572,248	11/1996	Allen et al.	395/330 X
5,613,056	3/1997	Gasper et al.	395/806 X
5,617,539	4/1997	Ludwig et al.	395/330 X
5,682,469	10/1997	Linnett et al.	345/473

OTHER PUBLICATIONS

Carlsson et al., "DIVE—a Multi–User Virtual Reality System", IEEE Virtual Reaity Annual International Symposium, pp. 394–400, Sep. 1993.

Ohya et al., "Real–time Reproduction of 3D Human Images in Virtual Space Teleconferencing", IEEE Virtual Reality Annual International Symposium, pp. 408–414, Sep. 1993.

Robinett, "Interactivity and Individual Viewpoint in Shared Virtual Worlds: The Big Screen vs. Networked Personal Displays", Computer Graphics, vol. 28, No. 2, pp. 127–130, May 1994.

Benford et al., "Supporting Cooperative Work in Virtual Environments", The Computer Journal, vol. 37, No. 8, pp. 653–668, 1994.

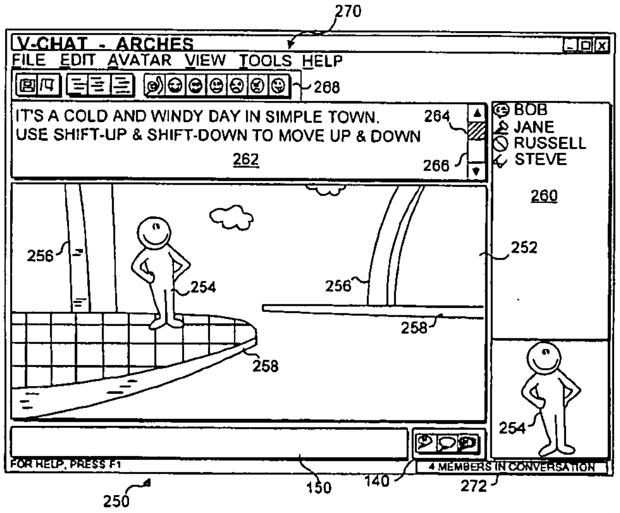
Fukuda, Kazutomo et al., "Hypermedia Personal Computer Communication System: Fujitsu Habitat," Fujitsu Sci. Tec. J., 26, 3, pp. 197–206.

Primary Examiner—Matthew M. Kim
Assistant Examiner—Crescelle N. dela Torre
Attorney, Agent, or Firm—Ronald M. Anderson

[57] **ABSTRACT**

Avatars representing participants in a graphic chat session are periodically animated to produce a gesture that conveys an emotion, action, or personality trait. Each participant in the chat session is enabled to select one of a plurality of different avatars to represent the participant in a graphic chat session. Associated with each avatar is a bitmap file that includes a plurality of frames illustrating the avatar in different poses, actions, and emotional states. Selected frames are displayed in rapid sequence in accord with a script file to create an animation effecting each gesture. The same script file is used to define a gesture for all of the avatars used in the chat session. A selected gesture can be transmitted with a text message to convey the user's emotional state. A gesture associated with the avatar is automatically displayed from time to time when the avatar is not otherwise gesturing or moving. The user can determine participants in the chat session with whom the user will interact, e.g., by defining a proximity radius around the user's avatar or by selecting the specific participants from a list. Avatars of participants that are outside the proximity radius (or otherwise not selected) and messages received from them are not displayed on the user's monitor.

31 Claims, 9 Drawing Sheets



IT'S A COLD AND WINDY DAY IN SIMPLE TOWN.
USE SHIFT-UP & SHIFT-DOWN TO MOVE UP & DOWN

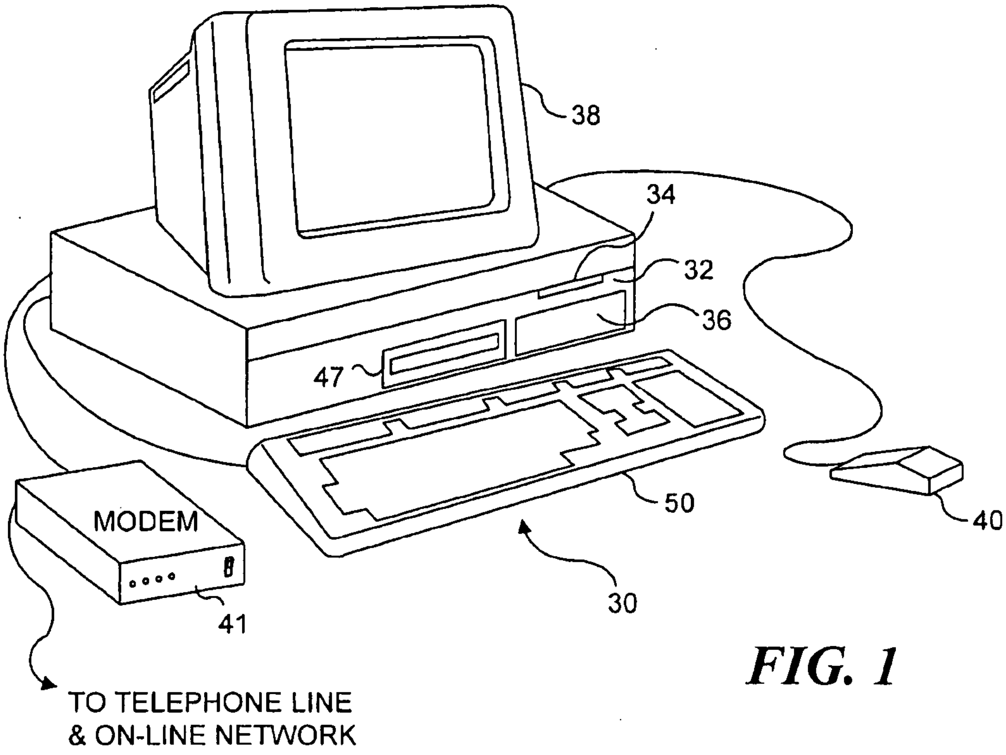

38
34
32
36
47
50
40
MODEM
41
30
TO TELEPHONE LINE
& ON-LINE NETWORK

FIG. 1

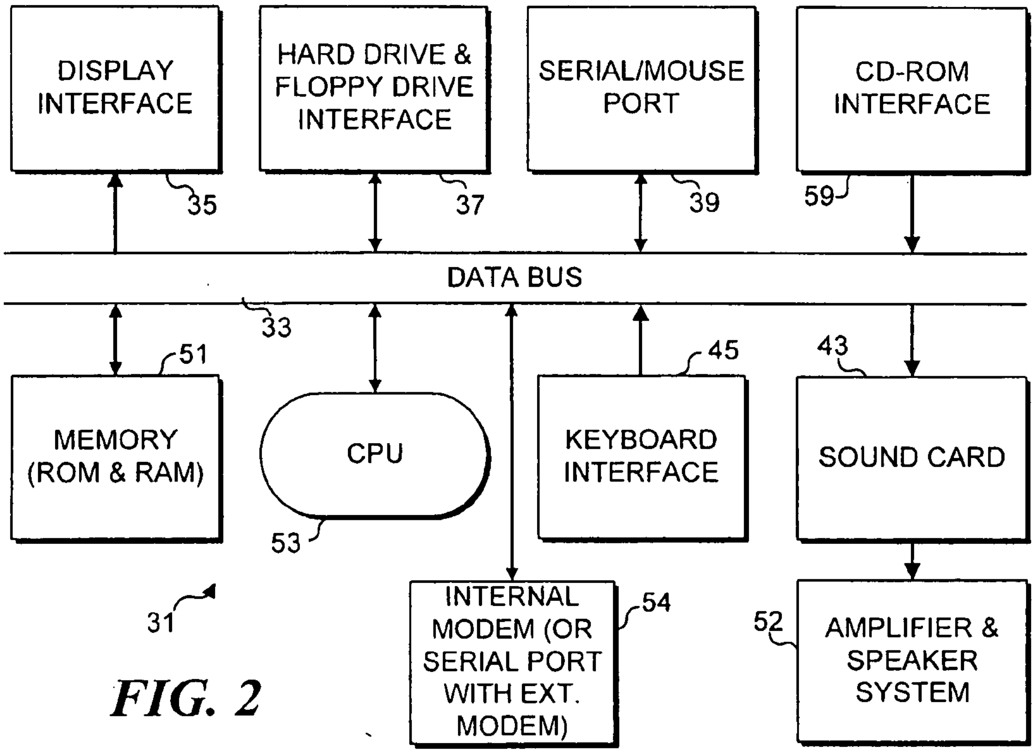

DISPLAY INTERFACE
35
HARD DRIVE & FLOPPY DRIVE INTERFACE
37
SERIAL/MOUSE PORT
39
CD-ROM INTERFACE
59
DATA BUS
33
51
MEMORY (ROM & RAM)
CPU
53
45
KEYBOARD INTERFACE
43
SOUND CARD
INTERNAL MODEM (OR SERIAL PORT WITH EXT. MODEM)
54
52
AMPLIFIER & SPEAKER SYSTEM
31

FIG. 2

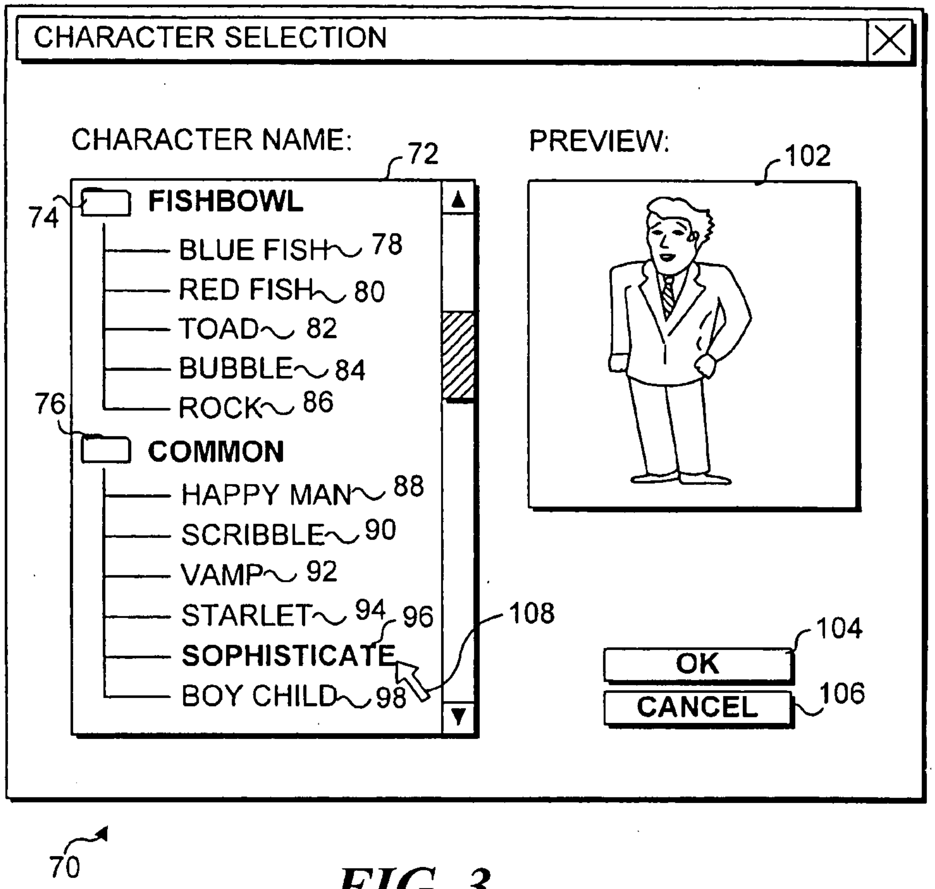


FIG. 3

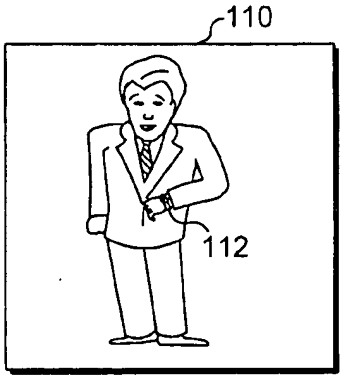


FIG. 4A

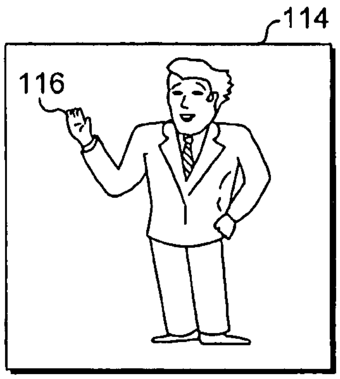


FIG. 4B

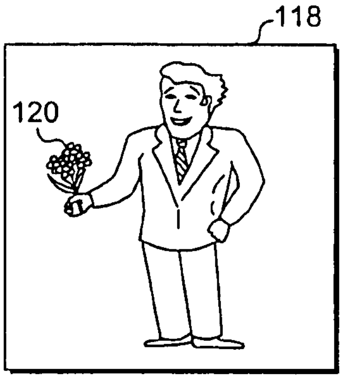


FIG. 4C

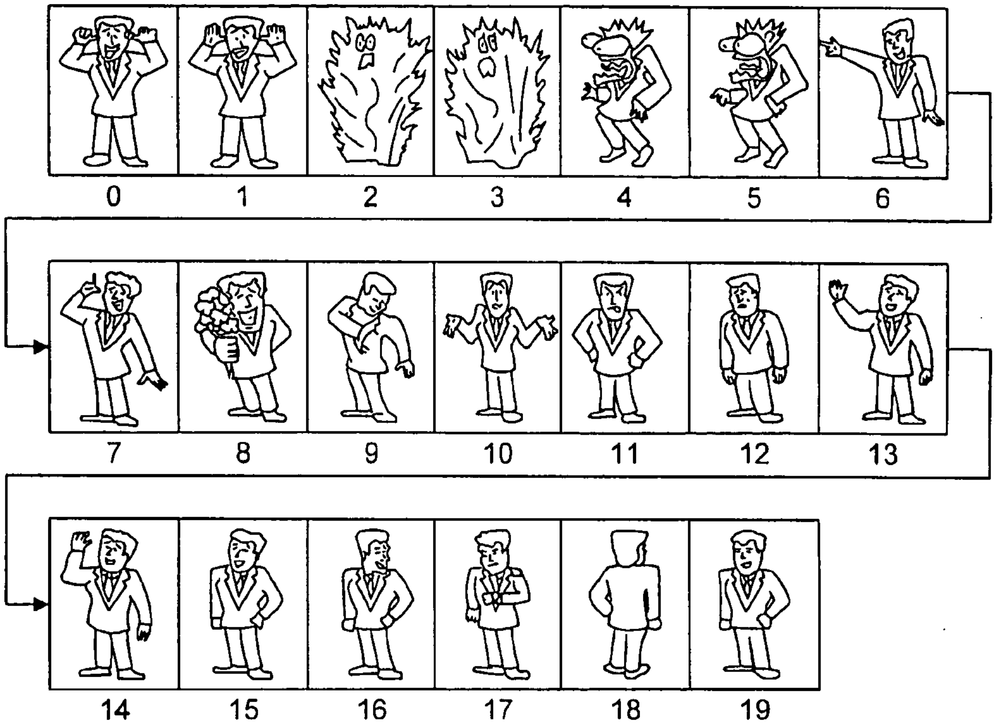

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19

FIG. 5

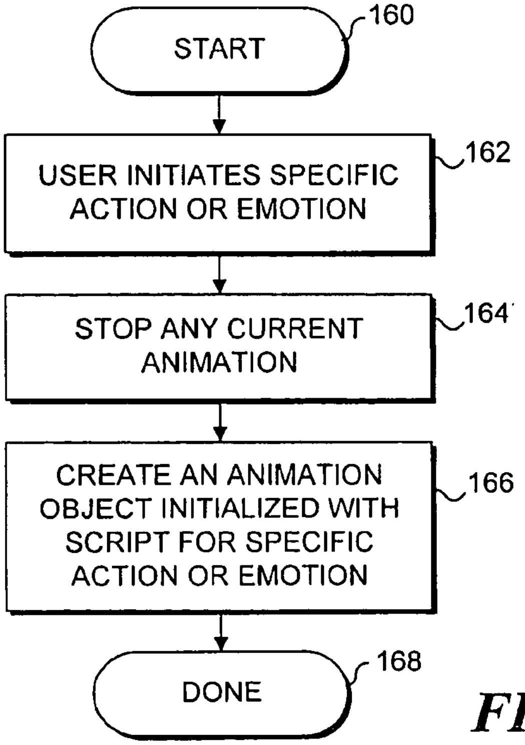

START
160
USER INITIATES SPECIFIC ACTION OR EMOTION
162
STOP ANY CURRENT ANIMATION
164
CREATE AN ANIMATION OBJECT INITIALIZED WITH SCRIPT FOR SPECIFIC ACTION OR EMOTION
166
DONE
168

FIG. 9

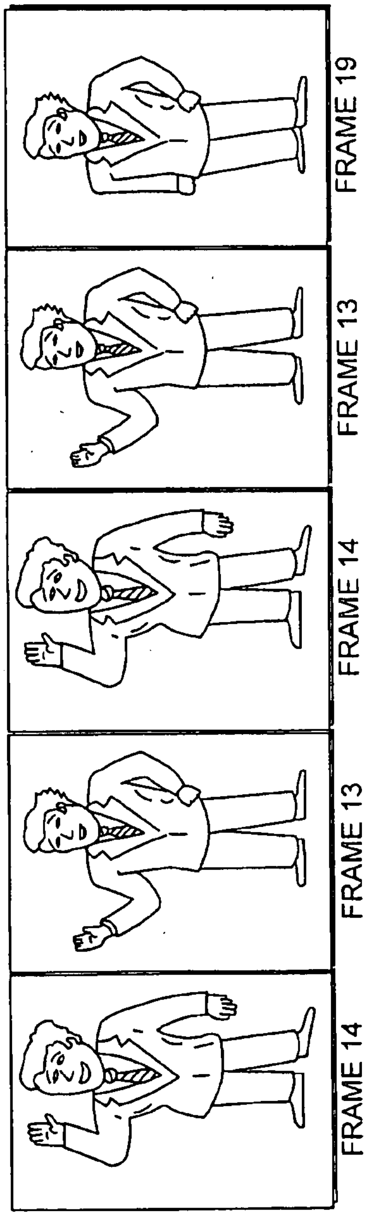


FIG. 6

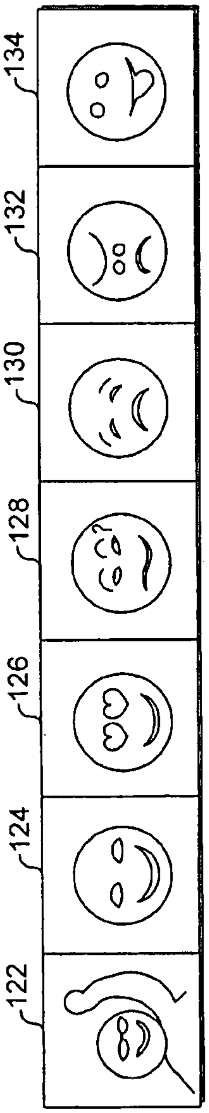


FIG. 7

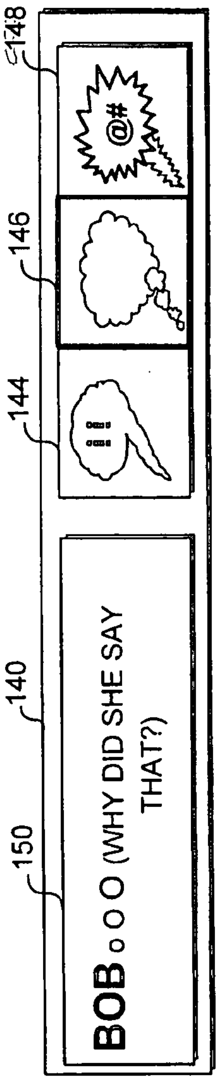


FIG. 8

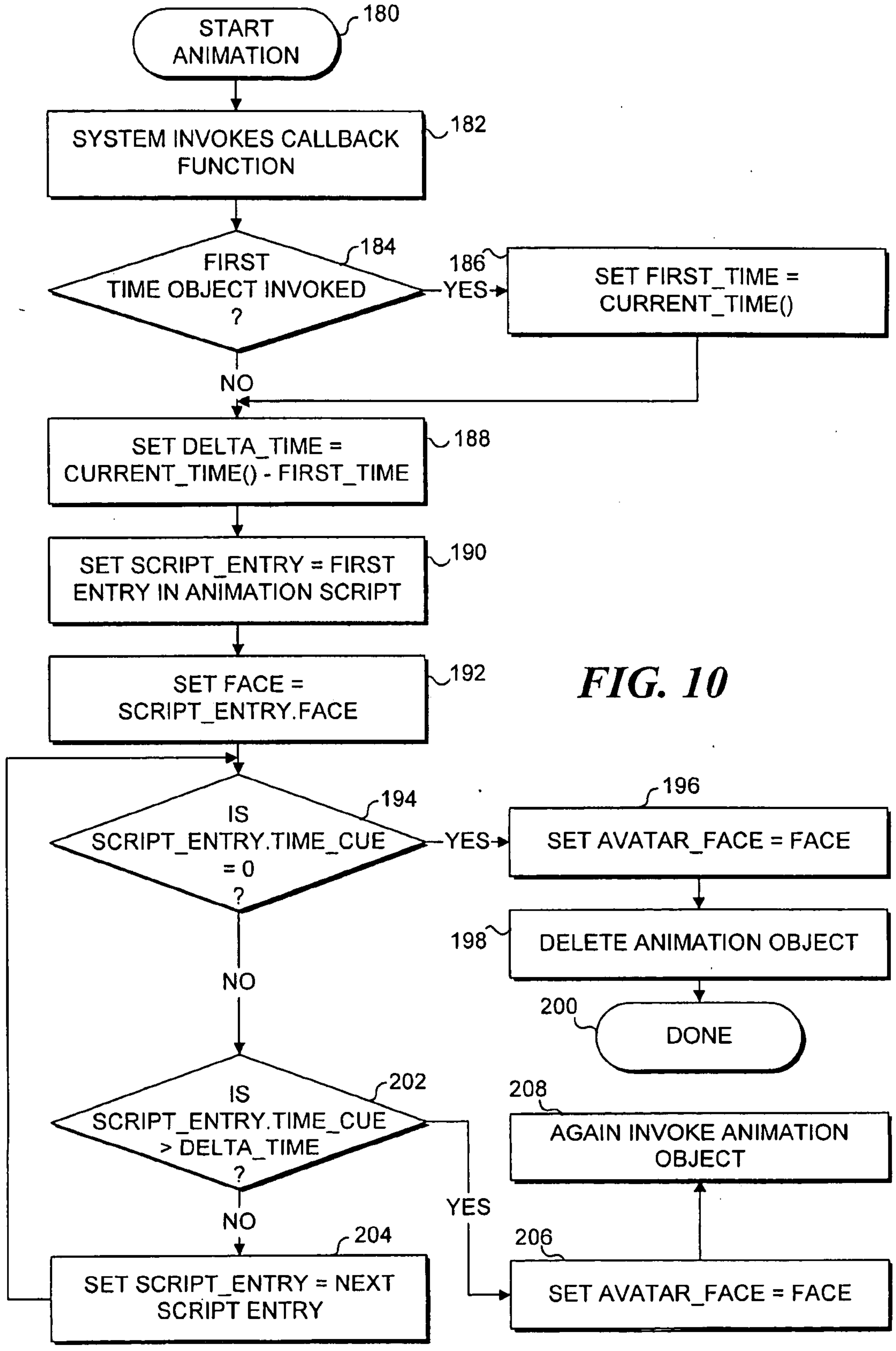

START ANIMATION
180
SYSTEM INVOKES CALLBACK FUNCTION
182
FIRST TIME OBJECT INVOKED ?
184
YES
186
SET FIRST_TIME = CURRENT_TIME()
NO
SET DELTA_TIME = CURRENT_TIME() - FIRST_TIME
188
SET SCRIPT_ENTRY = FIRST ENTRY IN ANIMATION SCRIPT
190
SET FACE = SCRIPT_ENTRY.FACE
192
IS SCRIPT_ENTRY.TIME_CUE = 0 ?
194
YES
196
SET AVATAR_FACE = FACE
198
DELETE ANIMATION OBJECT
200
DONE
NO
IS SCRIPT_ENTRY.TIME_CUE > DELTA_TIME ?
202
208
AGAIN INVOKE ANIMATION OBJECT
YES
NO
204
SET SCRIPT_ENTRY = NEXT SCRIPT ENTRY
206
SET AVATAR_FACE = FACE

FIG. 10

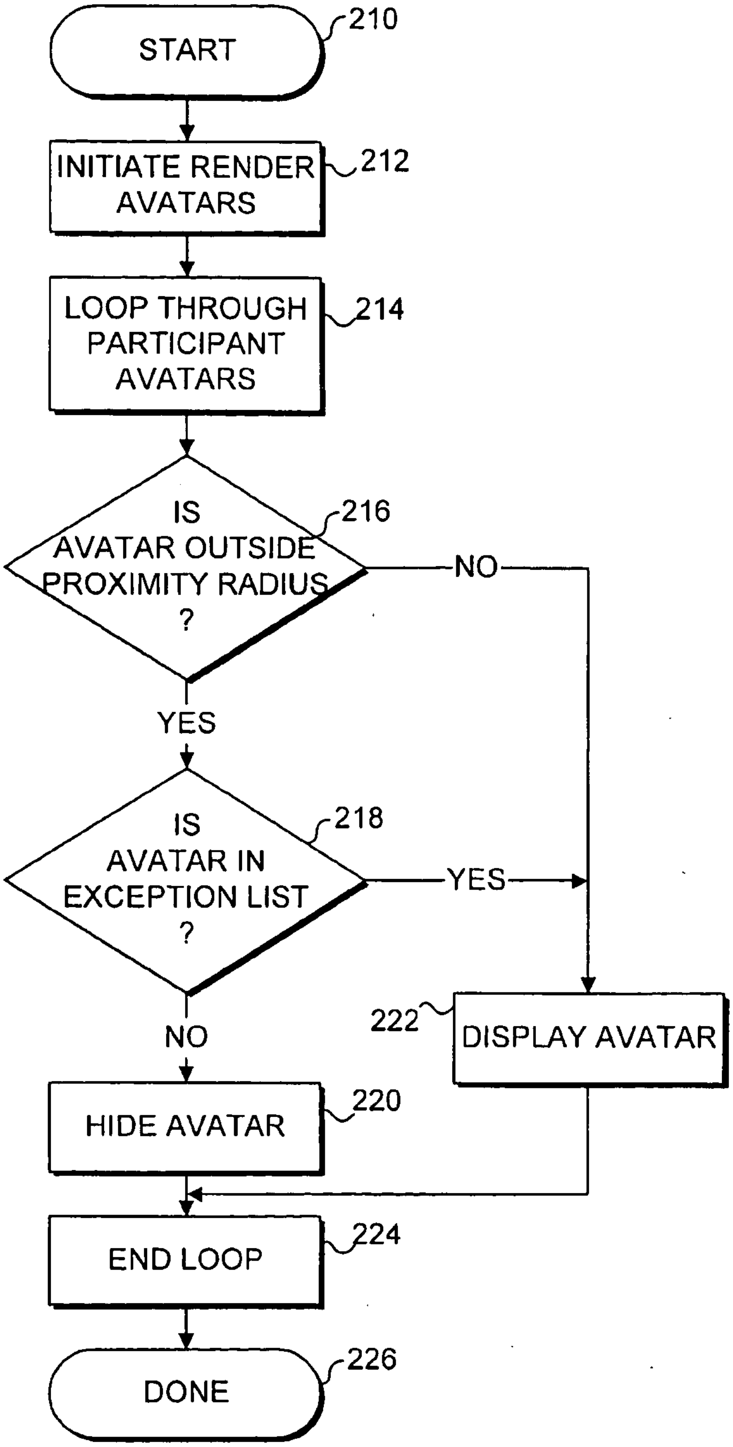

START
210
INITIATE RENDER AVATARS
212
LOOP THROUGH PARTICIPANT AVATARS
214
IS AVATAR OUTSIDE PROXIMITY RADIUS ?
216
NO
YES
IS AVATAR IN EXCEPTION LIST ?
218
YES
NO
DISPLAY AVATAR
222
HIDE AVATAR
220
END LOOP
224
DONE
226

FIG. 11

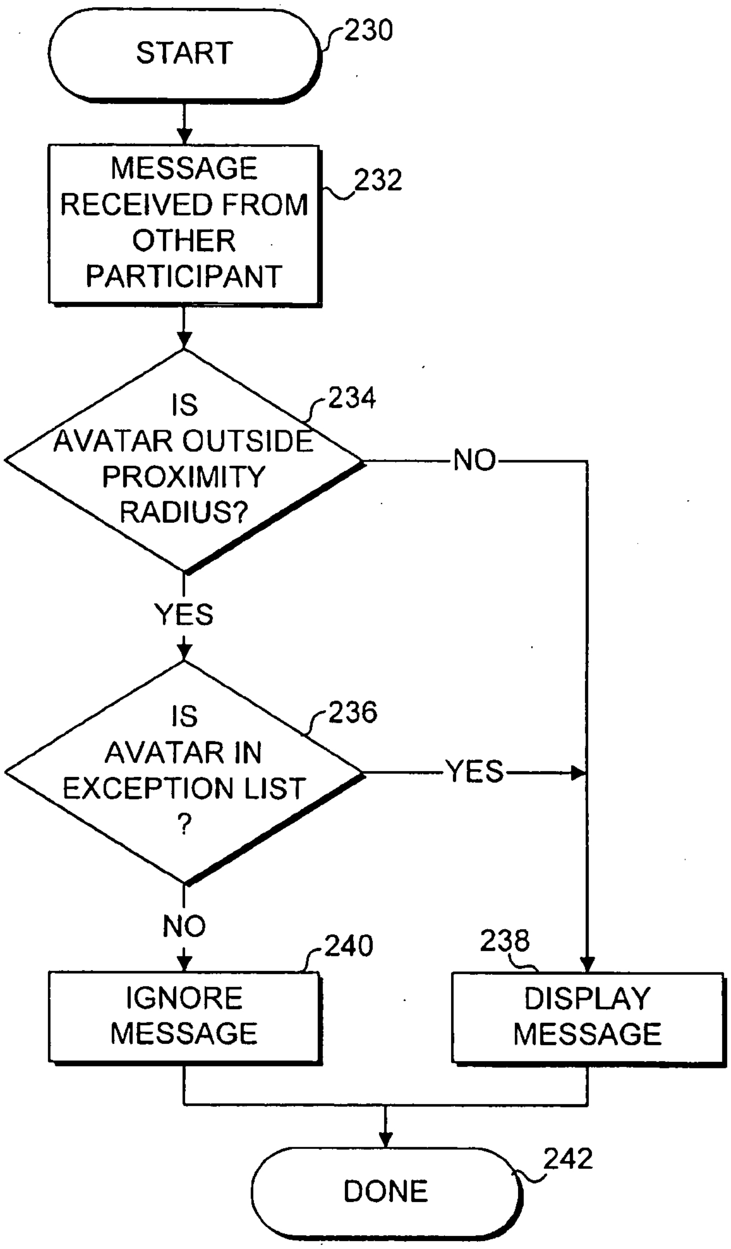
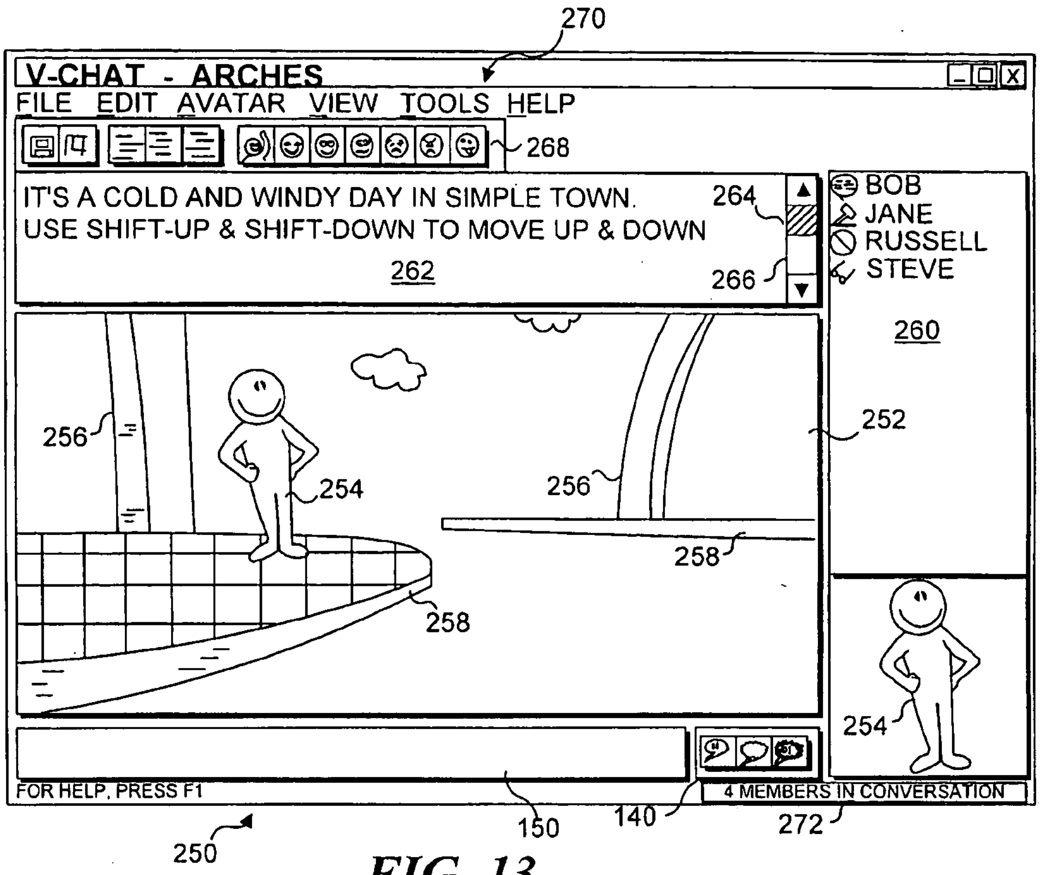
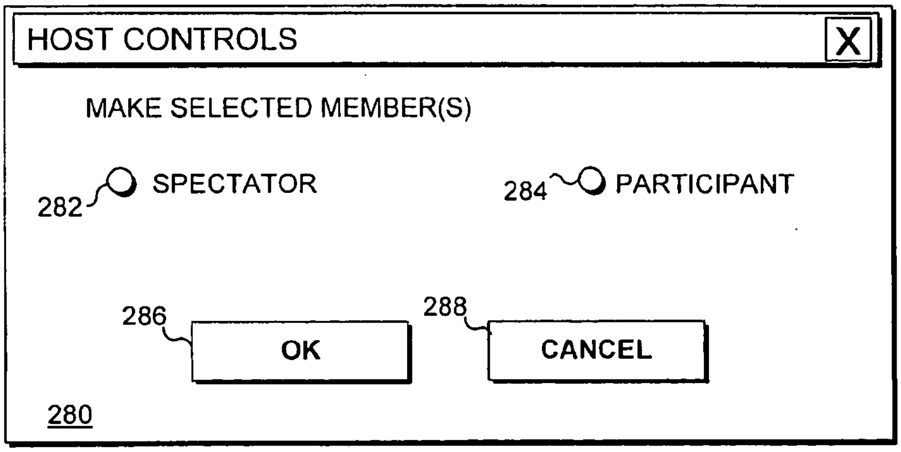

START
230
MESSAGE RECEIVED FROM OTHER PARTICIPANT
232
IS AVATAR OUTSIDE PROXIMITY RADIUS?
234
NO
YES
IS AVATAR IN EXCEPTION LIST ?
236
YES
NO
IGNORE MESSAGE
240
DISPLAY MESSAGE
238
DONE
242

FIG. 12



IT'S A COLD AND WINDY DAY IN SIMPLE TOWN.
USE SHIFT-UP & SHIFT-DOWN TO MOVE UP & DOWN

FIG. 13



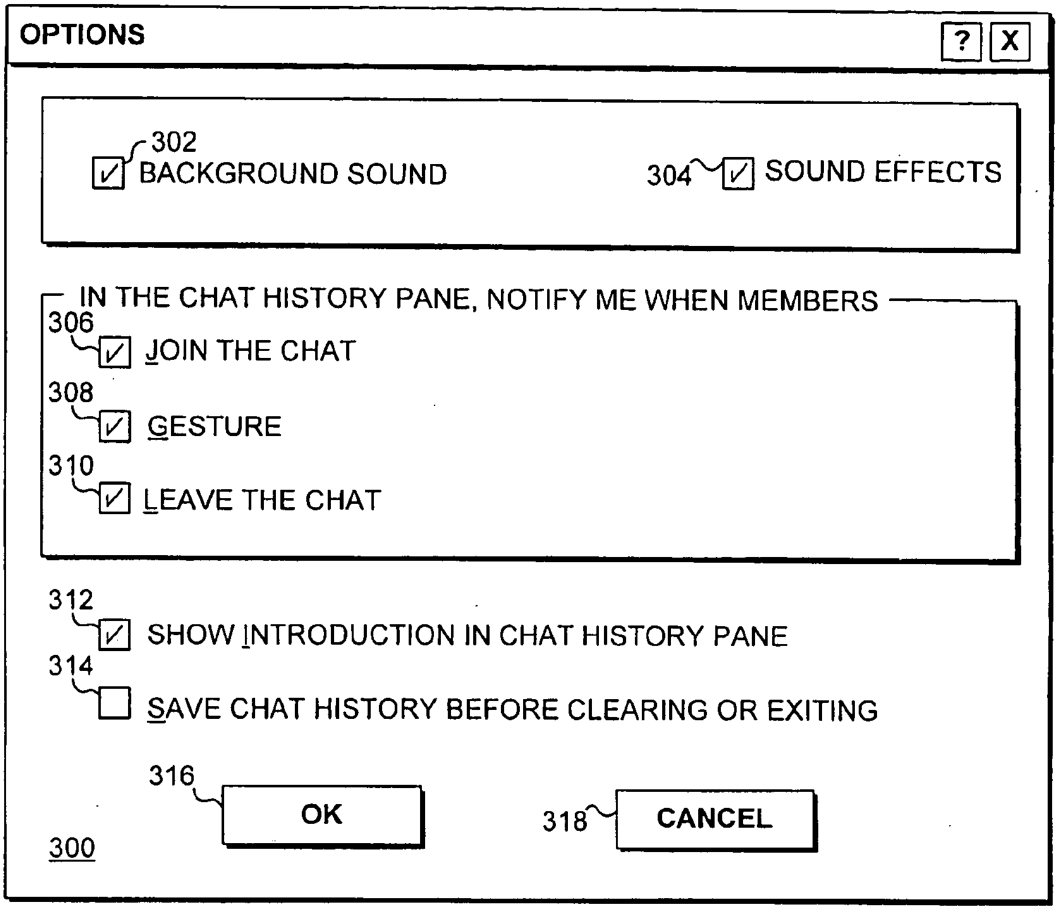
HOST CONTROLS

MAKE SELECTED MEMBER(S)

○ SPECTATOR ○ PARTICIPANT

OK CANCEL

FIG. 14



OPTIONS

☑ BACKGROUND SOUND

☑ SOUND EFFECTS

IN THE CHAT HISTORY PANE, NOTIFY ME WHEN MEMBERS

☑ JOIN THE CHAT

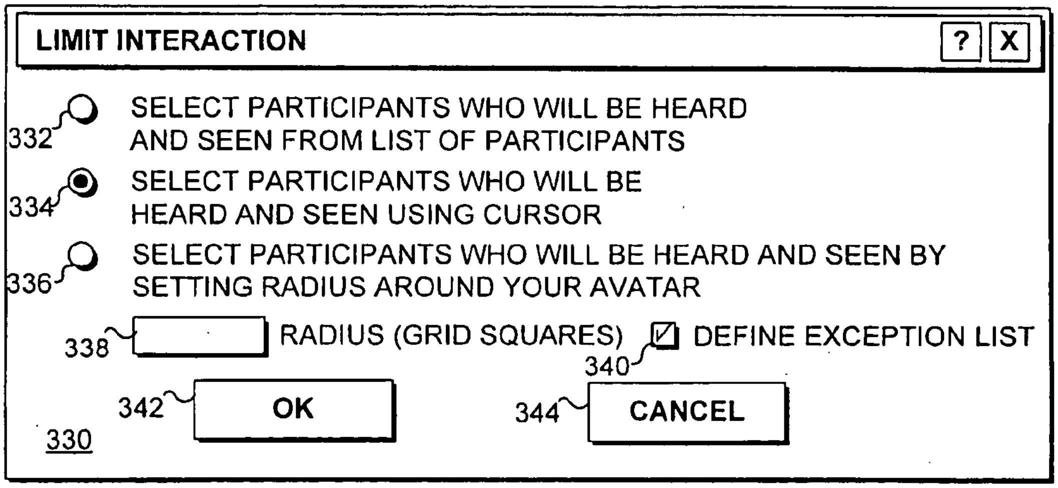
☑ GESTURE

☑ LEAVE THE CHAT

☑ SHOW INTRODUCTION IN CHAT HISTORY PANE

□ SAVE CHAT HISTORY BEFORE CLEARING OR EXITING

FIG. 15



LIMIT INTERACTION

○ SELECT PARTICIPANTS WHO WILL BE HEARD AND SEEN FROM LIST OF PARTICIPANTS

● SELECT PARTICIPANTS WHO WILL BE HEARD AND SEEN USING CURSOR

○ SELECT PARTICIPANTS WHO WILL BE HEARD AND SEEN BY SETTING RADIUS AROUND YOUR AVATAR

RADIUS (GRID SQUARES)

☑ DEFINE EXCEPTION LIST

FIG. 16

USE OF AVATARS WITH AUTOMATIC GESTURING AND BOUNDED INTERACTION IN ON-LINE CHAT SESSION

FIELD OF THE INVENTION

The present invention generally relates to the use of graphic representations of participants in a chat session, who are communicating using linked computers, and more specifically, it relates to the animation and interaction of avatars (graphic icons) representing the participants.

BACKGROUND OF THE INVENTION

Use of the computer for communicating on-line with others has recently become much more popular with the increased awareness by the public of the Internet and of services provided by commercial service networks. In addition to enabling access to information, exchange of e-mail messages, and downloading of files, a link to the Internet or to a commercial service network provides an individual with the opportunity to interact with others who are connected to the network.

One of the more common options for enabling several users of an on-line service to interact is through a chat session. A user joining a chat session is added to a list of participants and can then view comments transmitted by other participants and enter and transmit a response. In text only chat sessions, each user's screen is typically divided into two panes. Comments that have been transmitted by those participating in the chat session appear in one pane, and any message being entered by the user appears in the other pane on the user's computer display screen. For practical reasons, chat sessions are usually limited to a predefined number of participants. If any person attempts to join once the limit is reached, the person is typically notified that the chat session is full and precluded from joining. Alternatively, the person may be offered the opportunity to join another separate chat session on the same topic in which others are participating. In chat sessions involving a well known personality, hundreds of people may join the session, but only the host and the moderator are active in the chat session, and all others are simply observers. However, provision may be made to enable questions previously submitted by the observers to be displayed to solicit a response from the guest. The host controls the chat session. The virtual space in which each chat session occurs is sometimes referred to as a "room," since the participants interactively communicate just as if they were meeting in a room.

With the increasing use of modems operating at speeds of 28.8 Kbps on commercial networks, graphical chat sessions are becoming more practical. In a graphical chat session, all of the participants are represented by avatars or icons that are grouped in a graphic environment or "world." In addition to a graphic window showing the chat world, the display screen on each participant's computer typically still includes the chat pane and the message entry pane, as described above. When another user joins the chat session, the person's identifier, moniker, or name is added to a list of the participants, and an avatar for the new participant is added in the graphic chat world. The list normally appears in a third pane. When any participant leaves the chat session, the withdrawal is noted in the member pane, and the avatar representing the person is removed from the graphic chat world.

Although the graphic chat session provides visual information that improves a participant's knowledge of the other participants, the approach for displaying the participants in a conventional graphic chat world is 'somewhat stilted and artificial, because it fails to convey much information about the personality and emotional state of the participants as the chat session progresses. A graphic chat session of this type is implemented in the Worlds Chat paradigm, which was developed by World, Inc. In a Worlds Chat session, each avatar is associated with a plurality of bitmap sprites, each sprite representing the avatar from a different angle. The multiple views of each avatar do not provide any animation. A user can customize the sprites that represent him/her by modifying these bitmaps using a conventional paint program and a format conversion program that is provided by World, Inc.

In the ImagiNation Network (INN) developed by AT&T, users can customize their avatars by selecting various facial components such as the eyes, nose, and hair style from predefined options, in a manner much like that used in police identikits to create a likeness of a person. The avatars that are thus automatically produced blink their eyes at random times, but this limited animation fails to convey any emotion, action, or personality trait of the individual who is represented by the avatar.

Even though the particular avatar selected by a person and any customization applied with a paint program may reveal some of the individual's personality, such avatars are generally too static in nature and do not reflect the changing emotional state associated with the text messages transmitted by a participant in a graphic chat session. Ideally, a chat session in a virtual world should convey the same kinds of visual interaction that might occur in an actual face-to-face meeting of the people involved in a discussion, and the avatars representing the participants should thus clearly indicate the personalities of each individual and the emotions that are associated with the words communicated between the participants. Although present technology does not permit an ideal virtual world to be achieved during an on-line graphic chat session, it should be possible to animate the avatars sufficiently so that they can convey gestures that represent these traits. When people converse, gestures are an important facet of the communication, since they indicate the personality and emotional state of the speakers. In a graphic chat session, gestures can provide the same visual clues that they provide in a normal conversation. Although gestures are normally used in conversation without conscious thought, in a graphic chat session, it would be preferable for a participant to be able to select the gesture that will be used in combination with text that is transmitted to indicate the emotion or state of mind associated with the communication.

There are times when a participant in a chat session may wish to limit those with whom the person interacts. For example, if a discussion between two of the people involved in the chat session is of particular interest to a third party, the third person may not want to be distracted by communications transmitted from others in the chat session. In many cases, the participant may want to enable only selected persons in the chat session to view his/her avatar and the messages that are sent to those persons; however, this type of interactive control is currently not practical. Yet, it should be possible to selectively limit the group of participants with whom a person interacts so that only selected avatars in the chat session are seen by the person and so that only communications from the selected members of the group are observed by the person. Moreover, it would be preferable to select the members of the limited group that will be observed by the participant in a more graphical and natural manner.

When two people want to speak privately in a room, they simply move away from the others in the room so that their private conversation is not audible beyond the range of the other person with whom they are conversing. A similar approach should be applicable to limit those with whom a person interacts in a graphic chat world. Currently, no conventional graphic chat session provides a technique to spatially select the avatars of others that the participant wants to observe and from whom communications will be received. Providing this feature will enable a participant to perceive the avatars of those selected and to receive communications only from those members of the chat session who have been selected. The participant will not perceive the avatars or communications from those who are in the chat room, but were not selected.

SUMMARY OF THE INVENTION

In accord with the present invention, a method is defined for communicating a gesture by an avatar that represents a participant in an on-line graphic chat session. The method includes the step of providing an animation in which the avatar appears to move in a defined manner that conveys the gesture. During an idle period for the participant in the chat session, when the avatar is otherwise inactive, the animation is automatically initiated without requiring any input by the participant.

The method preferably also includes the step of providing the participant with a plurality of avatars. The participant is enabled to select the avatar that will represent the participant in the chat session.

The animation comprises a gesture chosen from a plurality of gestures, at least some of which are indicative of a personality trait and/or an emotion. Furthermore, the participant is enabled to select and initiate an animation employing the avatar, where the animation conveys a desired emotion and/or state of mind to another participant in the chat session. The animation selected by the participant to convey the desired emotion and/or state of mind may selectively be displayed in combination with a textual message that is transmitted by the participant.

Another aspect of the present invention is directed to a method for enabling a plurality of different gestures to be implemented by a plurality of different avatars that represent participants in an on-line graphic chat session. This method includes the step of providing a different script for each of the gestures. Each script is applicable to all of the plurality of avatars used in the chat session, and each gesture comprises a sequence of visual frames. The visual frames portray different views of an avatar to produce a visual impression of an animation of the avatar when rapidly displayed in the sequence. In the script for each gesture, specific visual frames comprising the sequence and time intervals determining a duration for displaying each visual frame of the sequence are indicated. The avatars representing participants in the chat session are caused to implement desired gestures by selectively executing the scripts for the desired gestures.

Yet another aspect of the present invention is directed to a system for use in enabling a participant in an on-line chat session who is represented by an avatar to indicate a personality trait and/or an emotion to others in the on-line chat session. The system includes an interface to a network on which the on-line chat session is being run; the interface enables the participant to transmit and receive data over the network. A display is provided for displaying a graphic representation of a virtual space in which the on-line chat session is occurring. Also included in the system is a memory for storing machine instructions, and a central processor for executing the machine instructions. The machine instructions, when executed by the central processor, cause the central processor to control the interface and the display. The central processor thus causes an animation of the avatar in the virtual space. This animation comprises a plurality of frames that are played in sequence so that the avatar appears to move in a defined manner within the virtual space. Movement by the avatar represents a gesture when viewed by others participating in the on-line chat session. During an idle period for the participant in the chat session, when the avatar representing the participant is otherwise inactive, the animation is automatically initiated.

Functions performed by the system in connection with the present invention are generally consistent with the steps of the methods described above.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

The foregoing aspects and many of the attendant advantages of this invention will become more readily appreciated as the same becomes better understood by reference to the following detailed description, when taken in conjunction with the accompanying drawings, wherein:

FIG. 1 is a schematic block diagram showing a personal computer and modem suitable for use in implementing the present invention;

FIG. 2 is a block diagram illustrating components of the personal computer that are included within its processor chassis;

FIG. 3 is a character selection dialog box enabling a user to select an avatar to represent the user in an on-line graphic chat session;

FIGS. 4A, 4B, and 4C respectively illustrate frames showing different gestures that are selectively executed by an exemplary avatar in accord with the present invention;

FIG. 5 illustrates a sequence of nineteen frames included in a bitmap file showing different views of an exemplary avatar;

FIG. 6 is a sequence of five frames that are played as defined by a script to produce a gesture animation of the exemplary avatar;

FIG. 7 is a dialog box illustrating a plurality of controls that can be selected by a user to control the animation of the avatar representing the user in an on-line chat session;

FIG. 8 is a text box in which the user enters a text message and defines the nature of the text message;

FIG. 9 is a flow chart that defines the logical steps implemented in displaying a gesture by executing an animation of an avatar;

FIG. 10 is a flow chart showing the logical steps followed to create an animation by following a predefined script;

FIG. 11 is a flow chart that illustrates the logical steps followed in selecting the persons that can communicate with a participant in the on-line chat session;

FIG. 12 is a flow chart that shows the steps for determining whether a message from another person is displayed to a participant;

FIG. 13 is a screen showing an example of an introductory virtual world or room displayed when a user joins a chat session;

FIG. 14 is a host control dialog box that is used by a monitor of the chat session to determine those who are participants and those who are spectators of a chat session;

FIG. 15 is an options dialog box that enables a user to select various options for the graphic chat sessions; and

FIG. 16 is a dialog box showing the options available to the user to limit interaction with other participants in the chat session.

DESCRIPTION OF THE PREFERRED EMBODIMENT

With reference to FIG. 1, a personal computer 30 is illustrated as an example of the type of computer typically used by a participant in a chat session in connection with the present invention. Although the personal computer is of the type intended to run Windows 95™, it is contemplated that other types of personal computers, such as those made by Apple Computer Corporation, will also be usable in executing software to implement the present invention. Personal computer 30 includes a processor chassis 32 in which is mounted a floppy disk drive 34, which is suitable for reading and writing data from and to a floppy disk (not shown), and a hard drive 36 suitable for nonvolatile storage of data and executable programs. A monitor 38 is included for displaying graphics and text produced when an executable program is being run on the personal computer and for use in connection with the present invention, for displaying a graphic chat session to a user.

Input can be provided to personal computer 30 using either a mouse 40 for manipulating a cursor (not shown) on monitor 38, which is used for selecting menu items and graphic controls displayed on the monitor by pressing an appropriate selection button (not shown) on the mouse, or by input entered by the user on a keyboard 50. Optionally, processor chassis 32 includes a CD-ROM drive 47, which is suitable for reading programs and data from a CD-ROM. To enable personal computer 30 to communicate during an on-line chat session, an external modem 41 is coupled to a serial port on processor chassis 32. Optionally, a modem may be included internally within processor chassis 32. The modem also connects to a telephone line to convey signals bi-directionally between computer 30 and a server at a remote on-line service (not shown) to which other participants in a chat session are connected in a similar fashion.

FIG. 2 shows a block diagram 31 in which components housed within processor chassis 32 are illustrated. A motherboard (not shown) includes a data bus 33, which provides bi-directional communication between these components and a CPU 53. The components include a display interface 35, which drives monitor 38, providing the video signals necessary to produce a graphic display during the chat session and when running other executable programs running on the personal computer. A hard drive and floppy drive interface 37 provides bi-directional communication between floppy drive 34 and hard drive 36, and data bus 33, enabling data and machine instructions comprising executable programs to be stored and later read into a memory 51. Memory 51 includes both a read only memory (ROM) and random access memory (RAM). The ROM is used for storing a basic input/output operating system (BIOS) used in booting up personal computer 30 and other instructions essential for its operation. Machine instructions comprising executable programs are loaded into the RAM via data bus 33 to control CPU 53.

A serial/mouse port 39 provides an interface for mouse 40 to data bus 33 so that signals indicative of movement of the mouse and actuation of the buttons on the mouse are input to CPU 53. An optional CD-ROM interface 59 couples optional CD-ROM drive 47 to data bus 33 and may comprise a small computer system interface (SCSI) or other appropriate type of interface designed to respond to the signals output from CD-ROM drive 47. Optionally, a sound card 43 is connected to data bus 33 and its output is coupled to an amplifier and speaker system 52 to provide a sound capability for personal computer 30. Output signals from keyboard 50 are connected to a keyboard interface 45, which conveys the signals from the keyboard to data bus 33. If external modem 41 is not used, an internal modem 54 can be provided, which is coupled directly to data bus 33. Alternatively, external modem 41 can be connected to the data bus through a serial port of personal computer 30.

It should be noted that instead of using a conventional modem, other types of digital adapters can be used to couple personal computer 30 to a telephone line. For example, an integrated services digital network (ISDN) would be a desirable alternative to the modem, since the ISDN interface can transfer data at a rate of 64 Kbps or more. At such a data transfer rate, there is very little delay in updating a screen on the monitor during a graphic chat session.

In connection with the present invention, it is contemplated that each participant in a graphic chat session will select a particular avatar to represent the person in the virtual world or room portrayed on monitor 38. Depending upon the subject matter of the chat session, a number of different, but appropriate, avatars will be provided from which a participant may make a selection. For example, if participating in a chat session involving gardening, a participant might select an avatar that appears as a gardener (male or female), a flower, a bee, a frog, a bird, or some other icon related to the subject. Moreover, as will be described below, a participant will have the opportunity to customize the avatar selected and alter the appearance of the avatar as used in various gestures or animations that can occur during a chat session.

The present invention provides the participant with a number of predefined avatars that can be selected to represent the individual in a chat session for a particular subject. This selection may be done when the participant is off-line, prior to making a connection with the service that runs the chat session. The software that enables the participant to select an avatar and to participate in a graphic chat session can either be downloaded from the service, or might be distributed on a floppy disk or CD-ROM disk. After the software is downloaded or transferred from the floppy disk or CD-ROM disk into personal computer 30, it can be executed by CPU 53, causing a dialog box to be displayed on monitor 38, so that the user can make a selection of the avatar for use in a graphic chat session. Alternatively, the selection may be made on-line, as appropriate for the subject matter of a graphic chat session that the user is joining.

FIG. 3 shows a character selection dialog box 70 that enables the user to select the avatar that will represent the user in an on-line chat session. Character selection dialog box 70 includes a block 72 in which folders 74 and 76 are displayed. Folder 74, which is entitled "Fishbowl," includes a plurality of bitmap files 78 through 86, each of which have names identifying various avatars associated with the fishbowl subject, including a Blue Fish 78, a Red Fish 80, a Toad 82, a Bubble 84, and a Rock 86. Similarly, folder 76 entitled "Common" includes a plurality of files 88 through 98 that identify additional avatars, including a Happy Man 88, a Scribble 90, a Vamp 92, a Starlet 94, a Sophisticate 96, and a Boy Child 98. Whereas the avatars provided in folder 74 might be appropriate for a chat session dealing with fish, those in folder 76 are more general in nature. Still other avatars can be selected by scrolling through the dialog box to display additional folders containing different groups of avatars.

In the example portrayed in FIG. 3, character selection dialog box **70** includes a cursor **108** that has been used to select Sophisticate **96** as the avatar that will represent a user. A preview box **102** illustrates the avatar associated with Sophisticate **96**. Controls **104** and **106** respectively enable the user to confirm (OK) the selection or Cancel it.

When connected to an on-line service and participating in the chat session, the avatar selected by the user in character selection dialog box **70** will appear in the virtual world or room with the avatars of the other participants. The virtual world is displayed in either a two-dimensional or three-dimensional mode. In addition, the user's identification or name will be added to the list of participants in the chat session. The user can move his/her avatar around within the room with mouse **40** and can turn the avatar to change the view of the other participants in the chat session that the user perceives on the monitor.

In addition to the view of the avatar appearing in preview box **102** when the avatar is selected, each avatar has a number of other different views that are employed in animations used to convey gestures. These gestures can be selected by the user for display to the other participants in a chat session in connection with a text message transmitted by the user. For example, in each of FIGS. **4A** through **4C**, the Sophisticate avatar is respectively shown in a different view used in three gestures. In a gesture represented by a view **110**, the avatar is checking a watch **112**; the gesture represented by view **110** might be selected to indicate that the user is impatient or to indicate that the user must leave the chat session because of an appointment. A gesture represented by a view **114** shows the avatar raising a hand **116** to wave at the other participants in the chat session; this gesture might be used in connection with greeting a friend who has just joined the graphic chat session. A gesture represented by a view **118** in FIG. 4C shows the Sophisticate avatar holding a bouquet of flowers **120** and might be employed when making an apology to a female participant for a harsh comment previously transmitted during the chat session. Alternatively, the gesture represented by view **118** could be used in connection with a flirtatious communication transmitted from the user represented by the Sophisticate avatar and directed to another participant.

Each of the avatars that are presented to a user for selection in character selection dialog box **70** corresponds to a different bitmap file. Each of the bitmap files contains a predefined number of frames that represent the avatar in different poses and/or emotional states. In FIG. **5**, frames **0** through **19** illustrate various views of the Sophisticate avatar. Since these frames are all included within a single bitmap file, the user is free to customize any frame in the bitmap file by opening the bitmap file within a paint program and modifying the expression on the avatar's face, the position of the avatar's limbs or other features, or any other aspect of the avatar, in any of the frames as desired. However, any of these frames may be employed in one or more of a plurality of gestures that are predefined. The frame numbers used in a predefined gesture are the same for all of the avatars employed in a chat session for a particular virtual world or room. Typically, several of the frames are displayed rapidly in sequence on a participant's monitor to produce an animation conveying a specific gesture. As is well known to those skilled in producing cartoon animations, the rapid display of a sequence of frames in which a figure is portrayed in slightly different poses causes the figure to appear to move in an animated fashion.

Each gesture is controlled by a script defined by the designer of the graphic chat session room. The designer

specifies all of the animation scripts that define the gestures that will be supported in that room. Each animation script for a gesture has the form:

```
Gesture name
    Time cue, avatar frame (from bitmap file)
    Time cue, avatar frame (from bitmap file)
    . . .
    End of gesture code
```

In each script file, the Gesture name is the name applied to the gesture as it appears to the user in a chat session interface. The Time cue is a time interval, which is measured in milliseconds, running from the time that the gesture is initiated on a participant's monitor. The Time cue parameter specifies the time that a specific avatar frame of the frames in the avatar bitmap file should be displayed. For example, the script for the "Greet" gesture is:

```
"Greet"
   1,    14
   500,  13
   1000, 14
   1500, 13
   2000, 19
   0,    0
```

In the above example, one millisecond after the Greet gesture is initiated, frame **14** is displayed, as shown in FIG. **6**. Five hundred milliseconds after the start of the gesture, frame **13** is displayed, followed in order by frame **14**, frame **13**, and frame **19**, each at 500 millisecond intervals. Frame **19** is the frame for the avatar that is used when an animation is not running, i.e., the "normal view" of the avatar. The **0,0** entry in the script file marks the end of the gesture. Other script files may include additional or fewer frames and may display the frames for different time intervals than indicated in this example. Since the frames are presented in a relatively rapid sequence to the user on monitor **38**, the gesture appears as an animation in which the avatar moves to convey the gesture and then assumes the normal position.

In the Greet gesture, the Sophisticate avatar waves his hand. If the user has selected a different avatar, the same numbered frames from the other avatar's bitmap file will be used when the script file for the selected gesture is executed. Therefore, all of the avatar bitmap files for a particular chat session space in this preferred embodiment have the same number of frames in a common order in the bitmap files, and each frame will typically depict a generally equivalent pose or emotion if the animation that is run when a gesture is selected is intended to correspond to the name of the gesture. Therefore, the bitmap files for all of the other avatars used in the chat space in which the Sophisticate avatar is used will also have twenty frames, with the nineteenth frame portraying the neutral or normal pose for each avatar.

However, different avatars may portray the same gesture in very different ways. For example, the nineteenth frame may portray the avatar as a man standing on his head or as a bird on a branch. The Greet gesture for a female avatar might portray the female avatar performing a curtsey, or blowing a kiss, rather than waving her hand. Thus, a user has considerable latitude in customizing the actions or animations performed by the avatar in response to each of the predefined scripts employed for a particular room of a chat session. So long as the user maintains the same number of frames in the bitmap file as are used for all of the avatars in the chat space, the artwork depicting the frames for each avatar can be modified by the user completely as desired. This option gives the user considerable latitude in customizing the avatar and the animations conveying the gestures implemented by the avatar. Once the bitmap file for a user's avatar is customized, it can be selectively published, i.e., uploaded to the server maintained by the service on which the chat session runs, so that other participants in a chat session can download the customized bitmap file into hard drives of their computers. If a participant in a chat session has not downloaded the customized bitmap file of the user, as when the user joins the chat session, the participant will see an amorphous ghost-like image that represents the user. Once the participant downloads the customized bitmap file for the avatar of the user, the user's customized avatar and gestures will be apparent to the participant.

In the preferred form of the present invention, each of the avatars has a gesture associated with it that is initiated at random times when the avatar is not otherwise moving or performing a different gesture. For example, the Sophisticate avatar shown in the preceding example may from time to time perform a Smile gesture. Although the preferred embodiment of the present invention provides for the designer of the chat space assigning a specific gesture to each avatar to be run during the idle times for the avatar, it is contemplated that the user be enabled to select this gesture that will run automatically. The gesture selected to run automatically for a particular avatar may then be chosen by the user to fit the user's personality and can be customized as described above. The automatically initiated gestures ensure that avatars in a chat session are animated, even if most of the participants in the chat session are not actively otherwise participating.

Various predefined gestures are provided in the script files for a typical graphic chat session, as indicated in FIG. 7. These animations are presented to a user in a gesture toolbar that includes gestures 122 through 134, and the user is given the option to select the gesture that will be initiated at random times. The graphic icons included on the toolbar are simply illustrative of the type of gesture and do not represent the appearance of the avatar selected by the user. The labels under the graphic icons indicate the personality trait, action, or emotion portrayed by the gesture, enabling a user to select a gesture based upon these label criteria. For example, if the user wishes to initiate a gesture 128 in response to a question to which the user does not know the answer, his avatar will follow a script that causes the avatar to be animated to produce a Shrug gesture. The details of the Shrug gesture may be very different for each avatar. The graphic icons on the controls used for initiating such gestures are sometimes referred to as "emoticons," since the gesture initiated by selecting one of these controls portrays an emotional state or personality trait of the user.

In the preferred embodiment, gestures are not embedded or associated with text messages that are transmitted by a participant for display to other participants. However, it is contemplated that a user will be enabled to select a gesture to accompany text that is transmitted for display to the other participants in the chat session. The gesture thus selected will provide emphasis of the user's emotional state in connection with the text message. Currently, in the preferred embodiment of the present invention, the user can select a gesture that indicates the user's emotional state in response to a prior communication within the chat session, for transmission without accompanying text, but a selected gesture and a text message can readily be transmitted together.

Textual messages transmitted to the other participants can indicate what the user is saying or thinking, and alternatively, can indicate a related action or emotional condition of the user. In FIG. 8, a text box 150 is provided to enable the user to compose a message to be transmitted to the other participants in the chat session. A "text balloon" control 144 can be selected by the user before transmitting the text typed into text box 150 to indicate that the text should be treated as a verbal communication. The other participants would see a message from Bob that reads, "Bob says hello," in response to a transmission from Bob of the text "hello," using text balloon control 144. Alternatively, as shown in the example, the user can select a "mental thought balloon" control 146 to indicate that the test represents the user's thoughts. As shown in text box 150, a user named Bob is thinking "Why did she say that?" The other participants would receive the message in the form shown in text box 150 and would understand that this message is in the form of a thought. In the third option, an emote control 148 can be selected to indicate that the text being transmitted represents an action or an emotional response. The other participants might receive text reading "Bob faints with laughter," in response to a message transmitted from Bob using emote control 148 that reads "faints with laughter." In the graphic chat room, it is contemplated that the text transmitted will be included within corresponding cartoon balloons like those represented on controls 144 through 148. Thus, the other participants in the chat session will understand the nature of the communication received from the user. Although not included in the current preferred embodiment, it is expected that such text messages will more clearly convey the user's current emotional state, if accompanied with an appropriate gesture effected by the user's avatar.

In FIG. 9, a flow chart illustrates the steps involved in performing a gesture using an animation defined by a script file. This procedure begins with a start block 160. In a block 162, a user initiates an animation object to produce a gesture to convey a specific action or emotion, for example, by selecting one of gesture emoticons 122 through 134. The selected gesture, as noted above, may optionally accompany a text message. In a block 164, any animation that was previously initiated is interrupted. Next, in a block 166, the animation object associated with the gesture selected by the user is initialized using the script for that gesture. Finally, the procedure terminates in a block 168 after the script and animation are concluded.

A flow chart shown in FIG. 10 includes the steps performed in creating an animation object in accord with block 166 of FIG. 9. The process starts in a block 180 and proceeds to a block 182 in which the system invokes a callback function. If the gesture was initiated automatically, the callback function is executed whenever the period of time randomly determined to apply between the previous automatic initiation of the gesture and the current initiation has elapsed. This function contains the code that performs the gesture's script by displaying the frames of the script in sequence to produce the animation. The callback function looks through the script for a script entry having a time cue that is greater than the interval of time elapsed since the animation object was first invoked. In a decision block 184, the procedure determines if this is the first time that this animation object was invoked (in the current instance). If so, a block 186 sets First_Time equal to Current_Time. Thereafter, or if this is not the first time that the animation object was invoked, a block 188 sets a parameter Delta_Time equal to the difference between Current_Time and First_Time. A block 190 then sets the parameter Script_Entry equal to the first entry in the animation script.

Next, in a block **192**, a parameter Face is set equal to the parameter Script_Entry.Face, which corresponds to the frame number specified in the script of the bitmap file for the avatar. A decision block **194** determines if a parameter Script_Entry.Time_Cue is equal to zero, which occurs at the end of the script file, and if so, a block **196** sets the parameter Avatar_Face equal to the (neutral or normal) Face, i.e., equal to frame **19** in the above example. Block **196** is thus implemented at the end of the script, restoring the avatar to its neutral frame state. A block **198** deletes the animation object, i.e., terminates its execution. Thereafter, a block **200** concludes the procedure.

However, if the parameter Script_Entry.Time_Cue is not equal to zero, the script is not yet finished, and the logic proceeds to a decision block **202**, which determines if the parameter Script_Entry.Time_Cue is greater than the Delta_Time. If not, the current script entry has already been executed and the procedure proceeds to a block **204** that sets the parameter Script_Entry equal to the next line in the script. The logic then loops back to decision block **194**. If the determination made in decision block **202** indicates that the parameter Script_Entry.Time_Cue is greater than the parameter Delta_Time, the procedure continues at a block **206**, which sets the Avatar_Face equal to Face, i.e., equal to the frame of the bitmap file previously designated in block **192**. A block **208** then again invokes the animation object, starting the procedure once more at block **180**, to select the next frame to be displayed, repeating until the condition in decision block **194** is met, which indicates that all entries of the script file have been processed. While the animation is being transmitted and displayed to the other participants in the chat session, it may be observed by the user on the monitor display screen, depending upon the view mode selected by the user. The user may selectively view his/her own avatar, or may view avatars of other participants in the chat session.

Another feature of the present invention enables a user to selectively determine if distant participants in the chat session will be hidden from the user. If this menu item is selected, the user can thus limit the participants in a chat session with whom the user will interact. In the preferred embodiment of the present invention, the host of the chat session determines the radius around each participant's avatar beyond which the avatars of other participants and the transmission from the other participants will not be evident to the user if the "hide distant members" (participants) menu option is selected by the user.

It is also contemplated that in subsequent preferred embodiments of the present invention, the user will be provided with further controls to limit the other participants and communications visible to the user. For example, the user can determine the participants with whom he/she will interact in a chat session by setting a proximity radius around his/her own avatar. Any avatars of other participants that are within the proximity radius will be "heard" and "seen" by the user. To determine the proximity radius, the user will select a menu item, causing a dialog box to be provided in which the user enters a nominal measure of the radius. Alternatively, the user may define the proximity radius using the mouse cursor to graphically encompass or surround the avatars of all those participants with whom the user selectively chooses to interact, i.e., to view their avatars and transmissions. The dialog box used for entering these options is discussed below. Alternatively, the user may directly select the participants whose avatars and communications will then be evident to the user, the avatars and communications of other participants being hidden from the user.

A flow chart in FIG. **11** includes the logical steps for implementing this aspect of the invention. Beginning with a start block **210**, the logic proceeds to a block **212** in which the rendering of the avatars on monitor **38** is initiated. In a block **214**, the program begins a loop that checks all of the participant avatars in the chat session. A decision block **216** determines if a specific avatar is outside a proximity radius selected by the user to limit the participants with whom the user will interact (or not in the group of avatars otherwise selected by the user). If the avatar is outside the proximity radius that was selected by the user, the logic proceeds to a decision block **218** to determine if the participant is in an exception list. In the current preferred embodiment, the exception list only includes the host for the chat session. However, it is contemplated that the exception list may also include the names (or other identification) of specific individuals with whom the user wants to interact in the current chat session. It is also possible to store the exception list for the user for use in subsequent chat sessions, so that avatar and communications of any participant in the exception list will always be evident to the user. A negative response to decision block **218** causes the avatar of the participant to be hidden from the user, as provided in a block **220**.

A negative response to decision block **216** leads to a block **222** in which the participant's avatar is displayed in the graphic chat world on the user's monitor. Similarly, if the participant is within the exception list referenced in decision block **218**, the avatar of that participant is also displayed on the user's monitor in block **222**. A block **224** ends the loop after all avatars have been checked in this manner. The procedure terminates in a block **226**.

Similarly, a flow chart shown in FIG. **12** is used to determine whether the user's monitor will display text messages that are received from other participants of the chat session. The logic starts in a block **230**, proceeding to a block **232** in which a message is received from one of the other participants in the chat session. A decision block **234** determines if the avatar for the participant from whom the message was received is outside the proximity radius determined by the user (or otherwise selected) to define the group of participants with whom the user wants to interact. If the avatar is outside the proximity radius (or not among those participants otherwise selected by the user), a decision block **236** determines if the avatar represents one of the participants that is in the exception list identifying participants with whom the user has chosen to always interact. If so, or if the avatar of the participant is inside the proximity radius (or among those participants otherwise selected by the user), a block **238** displays the message on the user's monitor that was received from the other participant. If not, a block **240** indicates that the message is ignored, so that the user does not see it displayed on his/her monitor. Following either blocks **238** or **240**, the logic concludes in a block **242**.

FIG. **13** illustrates a window **250** showing an example of an opening graphic chat session in a virtual world or room **252** called "ARCHES." As the user joins the graphic chat session taking place, he/she enters the virtual space looking at an avatar **254** representing the host of the chat session. Avatar **254** welcomes the user with an introductory text message shown in a history pane **262**.

In this example of a virtual world, a plurality of floating grids **258** support arches **256**. Participants in the chat session can move from grid to grid and can use the Shift-Up or Shift-Down key to move their avatars up or down relative to the floating grids. Messages that are transmitted to the user are displayed and scrolled in the history pane. Text that has scrolled out of view in the history pane can be accessed by the user by moving a scroll box **266** in a scroll bar **264** in the history pane.

The user can enter text to be transmitted to other participants in the chat session in text box **150**, as noted above. In addition, the nature of the text message that is prepared by the user can be identified by selecting the appropriate text bubble (and optionally choosing an appropriate gesture to be transmitted with the text by selecting one of the emoticons **140** from a toolbar **268**).

Other participants in the chat session are listed in a list box **260**. The user's name and symbol will be added to list box **260** after he/she receives the introductory message, typically with a notation indicating that the user has just joined the chat session. At the bottom right corner of the window, a block **272** indicates the number of participants in the chat session. A menu **270** provides for controlling the user's interaction in the chat session.

Each chat session is normally monitored by a host. The host has control of the chat session and is provided with controls such as shown in FIG. **14** in a dialog box **280**. In this dialog box, the host can indicate that one or more selected members are to be treated either as spectators or participants in the chat session, by choosing one of radio buttons **282** or **284**. Normally, those joining the chat session are enabled to participate. If the number of participants in the chat session reaches a predetermined limit, any latecomers will normally be precluded from joining. In some chat sessions involving a guest personality, only the host and the guest are participants, and all others in the chat session are spectators. Only participants can generate messages to be transmitted and initiate gestures implemented by their avatars. In contrast, anyone designated as a spectator can receive transmissions from the participants and can observe the avatars of the participants, but are not represented in the chat session by avatars and cannot transmit messages or initiate gestures.

Before or after joining the chat session, a user can set specific options in a dialog box **300** as shown in FIG. **15**. Check boxes **302** and **304** enable the user to indicate whether background sound and sound effects should be implemented. The user can also determine if he/she should be notified when members join the chat session, initiate a gesture, or leave the chat session by selecting check boxes **306**, **308**, and **310**, respectively. When first joining a chat session, selecting a check box **312** will cause the user to be presented with an introduction such as that shown in chat history pane **262** in FIG. **13**. Optionally, the chat history can be saved before the chat history pane is cleared or before the user exits the chat session by selecting a check box **314**. Once the user has indicated the appropriate options, selecting a button **316** labeled "OK" applies the selected options. Alternatively, selecting a button **318** cancels the selections.

When joining a chat session, the user is presented with a dialog box **330**, as shown in FIG. **16**, that enables the user to limit interaction with other participants in the chat session. Radio buttons **332**, **334**, and **336** enable the user to select one of three options to limit the interaction in this manner. Choosing radio button **332** enables the user to select the participants who will be heard and seen from the list of participants, like that shown in list box **260** in FIG. **13**. Choosing radio button **334** enables the user to choose participants with whom the user will interact using the cursor to select the avatars of those participants in the virtual world or room view. Choosing radio button **336** enables the user to select the partcipants by establishing the proximity radius around the user's avatar. If radio button **336** is selected, the user will be requested to enter the proximity radius in a text box **338**. When a check box **340** is selected, the user can define the exception list, which determines the participants with whom the user will always interact, regardless of the proximity radius entered or the participants selected from the list box for this chat session. A control button **342** applies the selections made in this dialog box, while a control button **344** cancels the selections.

Although the present invention has been described in connection with the preferred form of practicing it, those of ordinary skill in the art will understand that many modifications can be made thereto within the scope of the claims that follow. Accordingly, it is not intended that the scope of the invention in any way be limited by the above description, but instead be determined entirely by reference to the claims that follow.

The invention in which an exclusive right is claimed is defined by the following:

1. A method for communicating a gesture by an avatar that represents a participant in an on-line graphic chat session, comprising the steps of:

(a) providing an animation in which the avatar appears to move in a defined manner that conveys the gesture, said gesture being determined by the participant to convey at least one of a pluarily of different personality traits and/or current emotions;

(b) during an idle period for the participant in the chat session, when the avatar is otherwise inactive, automatically initiating the animation without requiring any input by the participant; and

(c) during an active period for the participant in the chat session, when the avatar is performing a selected action, automatically initiating another animation in which the avatar appears to move in a defined manner that conveys another gesture, the other gesture being determined by the participant to convey at least one of a plurality of different personality traits and/or current emotions.

2. The method of claim **1**, further comprising the steps of providing the participant with a plurality of avatars; and enabling the participant to select the avatar that will represent the participant in the on-line chat session.

3. The method of claim **1**, wherein the animation comprises gestures that are indicative of a personality trait and/or an emotion.

4. The method of claim **1**, further comprising the step of enabling the participant to select and initiate an animation employing the avatar, said animation conveying a desired emotion and/or state of mind to another participant in the chat session.

5. The method of claim **4**, wherein the animation selected by the participant to convey the desired emotion and/or state of mind is displayed simultaneously in combination with a textual message that is transmitted by the participant.

6. A method for enabling a plurality of different gestures to be implemented by a plurality of different avatars that represent participants in an on-line graphic chat session, comprising the steps of:

(a) providing a different script for each of the gestures, each script being applicable to all of the plurality of avatars used in the chat session, each gesture comprising a sequence of visual frames, said visual frames portraying different views of an avatar to produce a visual impression of an animation of said avatar when rapidly displayed in the sequence;

(b) in the script for each gesture, indicating specific visual frames comprising the sequence and time intervals determining a duration for displaying each visual frame of the sequence; and

(c) causing the avatars representing participants in the chat session to automatically implement desired gestures by selectively executing the scripts for the desired gestures, any change to a script to modify a desired gesture causing a corresponding change in said gesture for all of the different avatars executing said gesture.

7. The method of claim **6**, wherein each of the plurality of avatars has at least one gesture associated with it that is automatically initiated from time to time when the avatar is otherwise inactive during the on-line chat session.

8. The method of claim **7**, further comprising the step of enabling the participants in the chat session to select the gesture that is automatically initiated, each of said plurality of gestures indicating a different personality trait and/or emotion.

9. The method of claim **6**, further comprising the step of enabling a participant to select a personality trait and/or emotion that will be indicated by one of the gestures to determine the gesture that will be implemented by the avatar representing the participant.

10. The method of claim **6**, further comprising the step of enabling a participant to perceive communications from another participant in the chat session only if the other participant is represented by an avatar that is disposed within a defined distance of the participant's avatar.

11. The method of claim **6**, further comprising the step of enabling the participant to visually perceive a gesture implemented by the avatar that represents the participant in the chat session.

12. A system for use in enabling a participant in an on-line chat session who is represented by an avatar to indicate at least one of a plurality of different personality traits and/or current emotions to others in the on-line chat session, comprising:

(a) an interface to a network on which the on-line chat session is being run, said interface enabling the participant to transmit and receive data over the network;

(b) a display for displaying a graphic representation of a virtual space in which the on-line chat session is occurring;

(c) a memory for storing machine instructions; and

(d) a central processor for executing the machine instructions, said machine instructions, when executed by the central processor, causing the central processor to control the interface and the display, so that;

(i) an animation is provided for the avatar in the virtual space, said animation comprising a plurality of frames played in sequence so that the avatar appears to move in a defined manner within said virtual space, said movement by the avatar representing a gesture when viewed by others participating in the on-line chat session that conveys at least one of the plurality of different personality traits and/or current emotions;

(ii) during an idle period for the participant in the chat session, when the avatar representing said participant is otherwise inactive, the animation is automatically initiated; and

(iii) during an active period for the participant in the chat session, when the avatar representing the participant is performing a selected action, automatically initiating another animation in which the avatar appears to move in a defined manner that conveys another gesture, the other gesture being determined by the participant to convey at least one of a plurality of different personality traits and/or current emotions.

13. The system of claim **12**, wherein the machine instructions executed by the central processor further provide the participant with a plurality of avatars; and enable the participant to select the avatar that will represent the participant in the on-line chat session.

14. The system of claim **12**, wherein the machine instructions executed by the central processor further enables the participant to select the animation that is initiated during the idle period from a plurality of different animations, each animation indicating a different personality trait and/or emotion.

15. The system of claim **12**, wherein the machine instructions executed by the central processor further enable the participant to selectively initiate an animation that conveys a desired emotion and/or state of mind of the participant to another participant in the chat session.

16. The system of claim **15**, wherein the animation selected by the participant to convey the desired emotion and/or state of mind of the participant is simultaneously activated in combination with a textual message that is transmitted by the participant.

17. A system for enabling a plurality of different gestures to be implemented by a plurality of different avatars that represent participants in an on-line graphic chat session, comprising:

(a) an interface to a network on which the chat session is being run, said interface enabling a participant to transmit and receive data over the network;

(b) a display for displaying a graphic representation of a virtual space in which the chat session is occurring;

(c) a memory for storing machine instructions; and

(d) a central processor for executing the machine instructions, said machine instructions, when executed by the central processor, causing the central processor to control the interface and the display, so that:

(i) a different script is provided for each of the gestures, each script being applicable to all of the plurality of avatars used in the chat session, each gesture comprising a sequence of visual frames, said visual frames portraying different views of an avatar to produce a visual impression of an animation of said avatar when rapidly displayed on the display in the sequence;

(ii) in the script for each gesture, specific visual frames comprising the sequence and time intervals determining a duration for displaying each visual frame of the sequence are indicated; and

(iii) the avatars representing participants in the chat session are caused to automatically implement desired gestures by selectively executing the scripts for the desired gestures, any change to a script to modify a desired gesture causing a corresponding change in said gesture for all of the different avatars executing said gesture.

18. The system of claim **17**, wherein the avatar has associated with it a script that determines a sequence of the visual frames employed to produce a gesture that is automatically initiated when the avatar is otherwise idle.

19. The system of claim **17**, wherein each avatar is associated with a graphic file comprising a predefined number of visual frames, said script referencing specific frames in the graphic file so that different views of the avatar indicated by the script are displayed when the script is executed to implement a gesture.

20. A method for enabling a participant in a graphic on-line chat session who is represented by an avatar to restrict communication with others participating in the on-line chat session, comprising the steps of:

(a) providing the participant with an identification of other persons currently participating in the on-line chat session;

(b) enabling the participant to select specific persons that are currently participating in the on-line chat session from whom the participant will perceive communications during the on-line chat session; and

(c) precluding the participant from perceiving communications from other than the selected specific persons, during the on-line chat session.

21. The method of claim 20, wherein the step of enabling the participant to select the specific persons comprising the step of providing a user interface tool that enables the participant to indicate a defined space adjacent to the avatar that represents the participant, so that only communications from any of the other persons participating in the on-line chat session who are represented by an avatar disposed within said defined space will be perceived by the participant.

22. The method of claim 21, wherein the participant is provided with a graphic control to set a radius around the avatar representing the participant to define said space.

23. The method of claim 20, wherein the step of enabling the participant to select specific persons that are currently participating in the on-line chat session comprises the step of enabling the participant to select the avatars representing any of the other persons who are participating in the on-line chat session, using a graphic pointing device.

24. The method of claim 20, wherein the step of enabling the participant to select specific persons that are currently participating in the on-line chat session comprises the step of enabling the participant to select the specific persons from a list of participants in the on-line chat session.

25. The method of claim 20, wherein the participant selects the specific persons by using a pointing device to trace a path defining a perimeter of a defined space in which the avatars representing the specific persons are disposed.

26. A method for enabling a participant in a graphic on-line chat session who is represented by an avatar to restrict communication with others participating in the on-line chat session, comprising the steps of:

(a) providing the participant with an identification of other persons participating in the on-line chat session;

(b) enabling the participant to select specific persons from whom the participant will perceive communications during the on-line chat session, the participant employing a graphic control to set a radius around the avatar representing the participant to indicate a defined space adjacent to the avatar, so that only communications from any of the other persons participating in the on-line chat session who are represented by an avatar disposed within said defined space will be perceived by the participant; and

(c) precluding the participant from perceiving communications from other than said specific persons, during the on-line chat session.

27. A method for enabling a participant in a graphic on-line chat session who is represented by an avatar to restrict communication with others participating in the on-line chat session, comprising the steps of:

(a) providing the participant with an identification of other persons participating in the on-line chat session;

(b) enabling the participant to select specific persons from whom the participant will perceive communications during the on-line chat session, the participant using a pointing device to trace a path defining a perimeter of a defined space in which the avatars representing the specific persons are disposed; and

(c) precluding the participant from perceiving communications from any person represented by an avatar that is disposed outside said perimeter, during the on-line chat session.

28. A method for communicating a gesture by an avatar that represents a participant in an on-line graphic chat session, comprising the steps of:

(a) providing an animation in which the avatar appears to move in a defined manner that conveys the gesture, said gesture being determined by the participant;

(b) during an idle period for the participant in the chat session, when the avatar is otherwise inactive, automatically initiating the animation without requiring any input by the participant; and

(c) enabling the participant to perceive communications from another participant in the chat session only if the other participant is represented by an avatar that is disposed within a defined distance of the participant's avatar.

29. A method for enabling a plurality of different gestures to be implemented by a plurality of different avatars that represent participants in an on-line graphic chat session, comprising the steps of:

(a) providing a different script for each of the gestures, each script being applicable to all of the plurality of avatars used in the chat session, each gesture comprising a sequence of visual frames, said visual frames portraying different views of an avatar to produce a visual impression of an animation of said avatar when rapidly displayed in the sequence;

(b) in the script for each gesture, indicating specific visual frames comprising the sequence and time intervals determining a duration for displaying each visual frame of the sequence;

(c) causing the avatars representing participants in the chat session to implement desired gestures by selectively executing the scripts for the desired gestures; and

(d) enabling a participant to automatically perceive communications from another participant in the chat session only if the other participant is represented by an avatar that is disposed within a defined distance of the participant's avatar.

30. A system for use in enabling a participant in an on-line chat session who is represented by an avatar to indicate a personality trait and/or an emotion to others in the on-line chat session, comprising:

(a) an interface to a network on which the on-line cat session is being run, said interface enabling the participant to transmit and receive data over the network;

(b) a display for displaying a graphic representation of a virtual space in which the on-line chat session is occurring;

(c) a memory for storing machine instructions; and

(d) a central processor for executing the machine instructions, said machine instructions, when executed by the central processor, causing the central processor to control the interface and the display, so that;

(i) an animation is provided for the avatar in the virtual space, said animation comprising a plurality of frames played in sequence so that the avatar appears to move in a defined manner within said virtual space, said movement by the avatar representing a gesture when viewed by others participating in the on-line chat session;

(ii) during an idle period for the participant in the chat session, when the avatar representing said participant is otherwise inactive, the animation is automatically initiated; and

(iii) the participant is enabled to perceive communications from another participant in the chat session only if the other participant is represented by an avatar that is disposed within a defined distance of the participant's avatar.

31. A method for enabling a participant in a graphic on-line chat session who is represented by an avatar to restrict communication with others participating in the on-line chat session, comprising the steps of:

(a) providing the participant with an identification of other persons participating in the on-line chat session; and

(b) enabling the participant to select specific persons from whom the participant will perceive communications during the on-line chat session by employing a user interface tool to indicate a defined space adjacent to the avatar that represents the participant so that the participant only perceives communications from the specific persons participating in the on-line chat session who are represented by an avatar disposed within the defined space.

* * * * *

US 20020026483A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2002/0026483 A1**

Isaacs et al. (43) **Pub. Date: Feb. 28, 2002**

(54) **SYSTEM, METHOD AND APPARATUS FOR COMMUNICATING VIA INSTANT MESSAGING**

(76) Inventors: **Ellen Isaacs**, Belmont, CA (US); **Dipti Ranganathan**, Dallas, TX (US); **Alan Walendowski**, Belmont, CA (US)

Correspondence Address:
Samuel H. Dworetsky
AT&T CORP.
P.O. Box 4110
Middletown, NJ 07748-4110 (US)

(21) Appl. No.: **09/823,167**

(22) Filed: **Mar. 30, 2001**

Related U.S. Application Data

(63) Continuation-in-part of application No. 09/609,893, filed on Jul. 5, 2000. Non-provisional of provisional application No. 60/260,035, filed on Jan. 5, 2001. Non-provisional of provisional application No. 60/264,421, filed on Jan. 26, 2001. Non-provisional of provisional application No. 60/184,180, filed on Feb. 22, 2000.

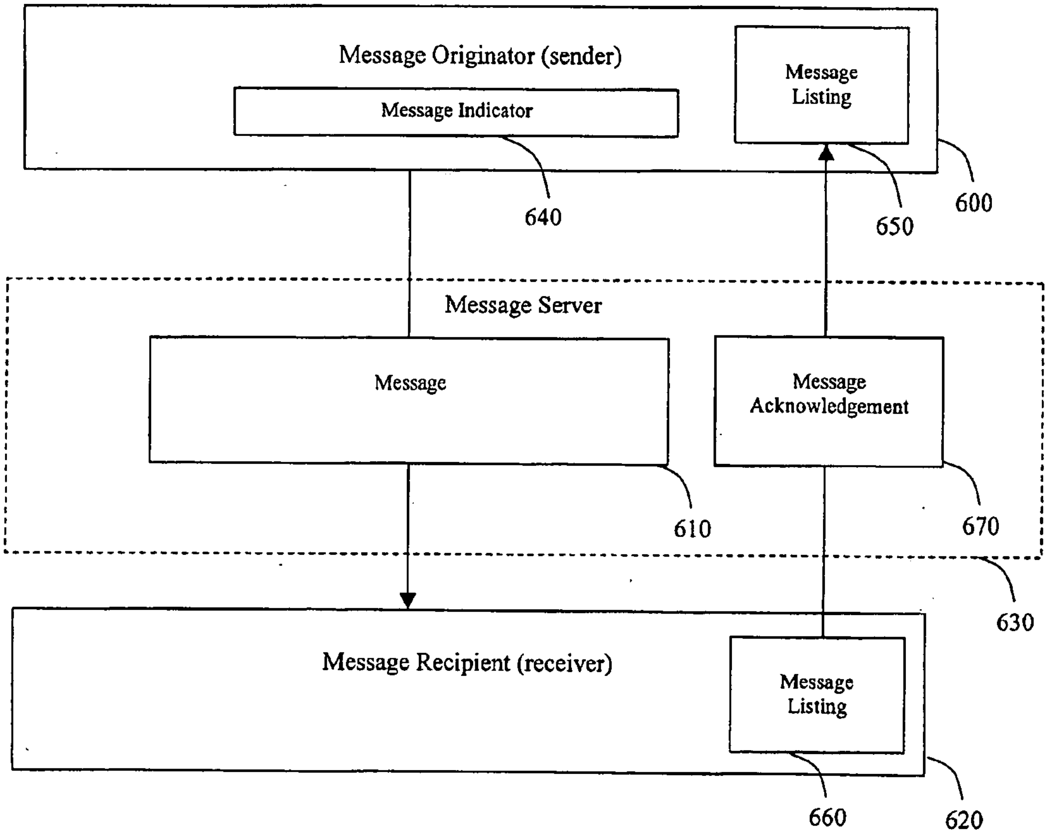
Publication Classification

(51) **Int. Cl.**^{7} **G06F 15/16**; G09G 5/00

(52) **U.S. Cl.** **709/206**; 345/751; 345/864

(57) **ABSTRACT**

A system, method and apparatus for facilitating communication among a number of distributed clients in a distributed network is disclosed. A user, such as through a personal digital assistant device, may select one or more instant messages for transmission to one or more other users in the network. The instant messages may be sound instant message and/or text instant messages. During messaging, message status indicators provide users with the status of their respective messages. In one embodiment, the messages may be deemed to be either pending or received as distinguished by a pending status indicator and a received status indicator.



AT&T IDS No. 2000-0027B-CIP

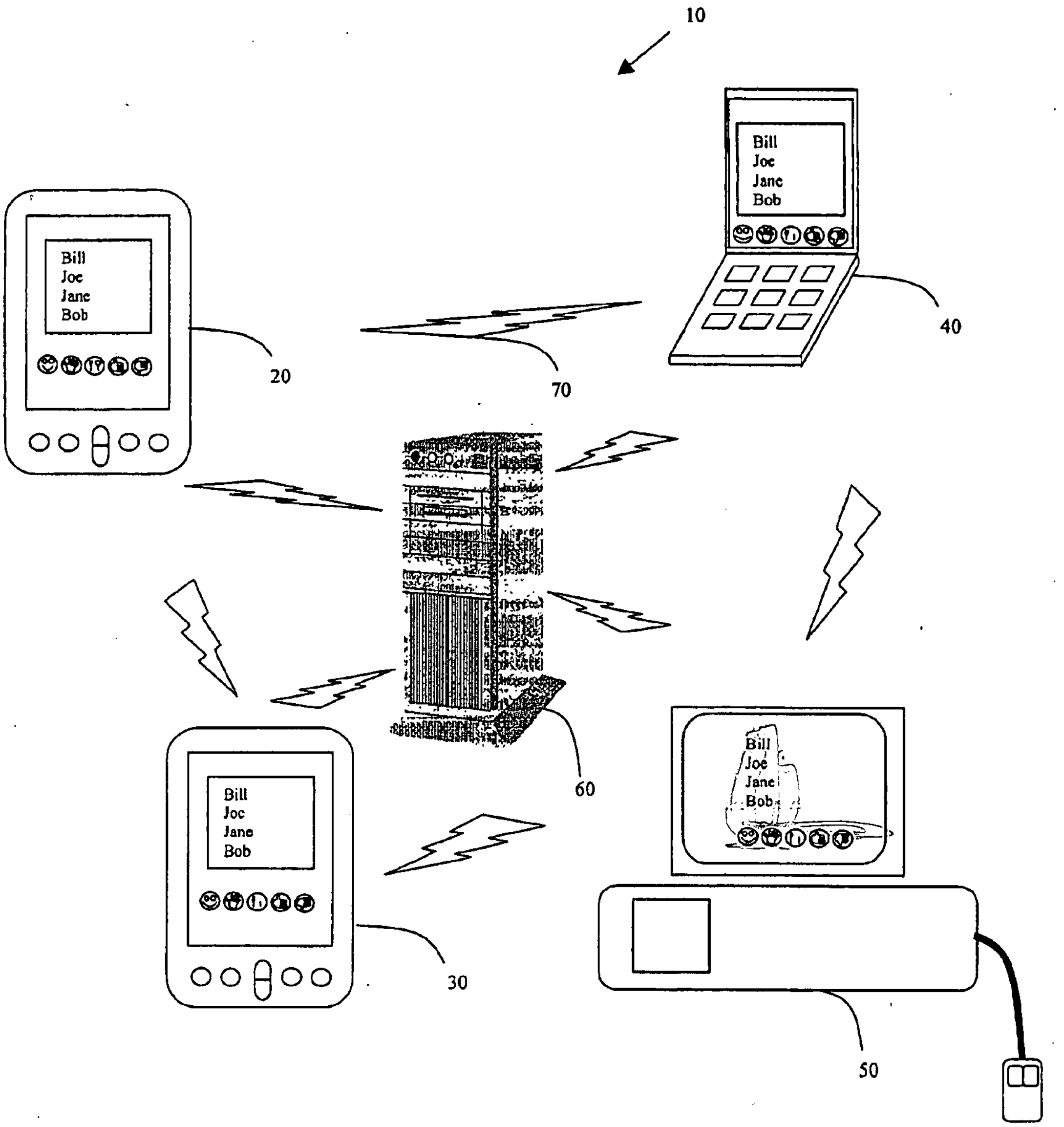


FIG. 1

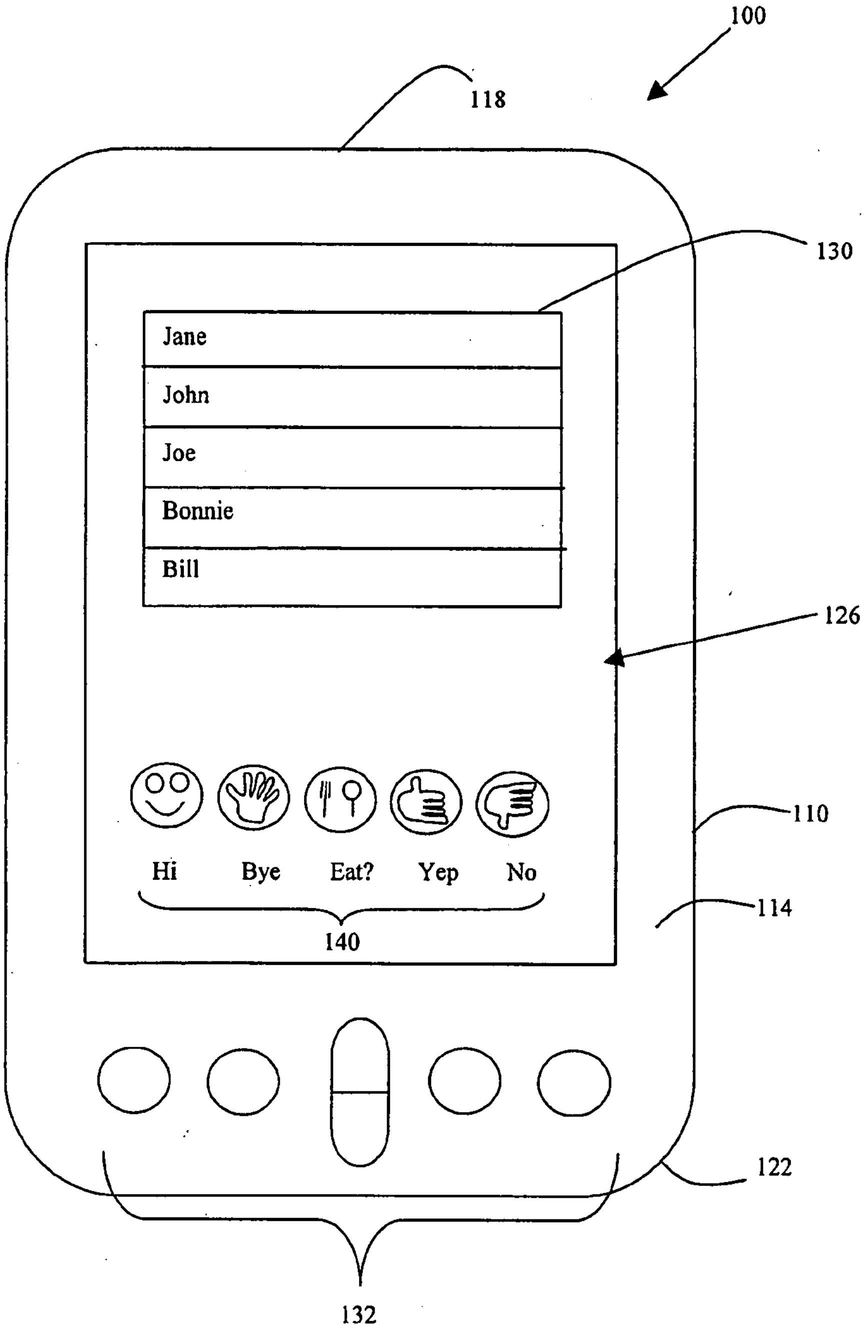

100
118
130
Jane
John
Joe
Bonnie
Bill
126
Hi
Bye
Eat?
Yep
No
140
110
114
122
132

FIG. 2

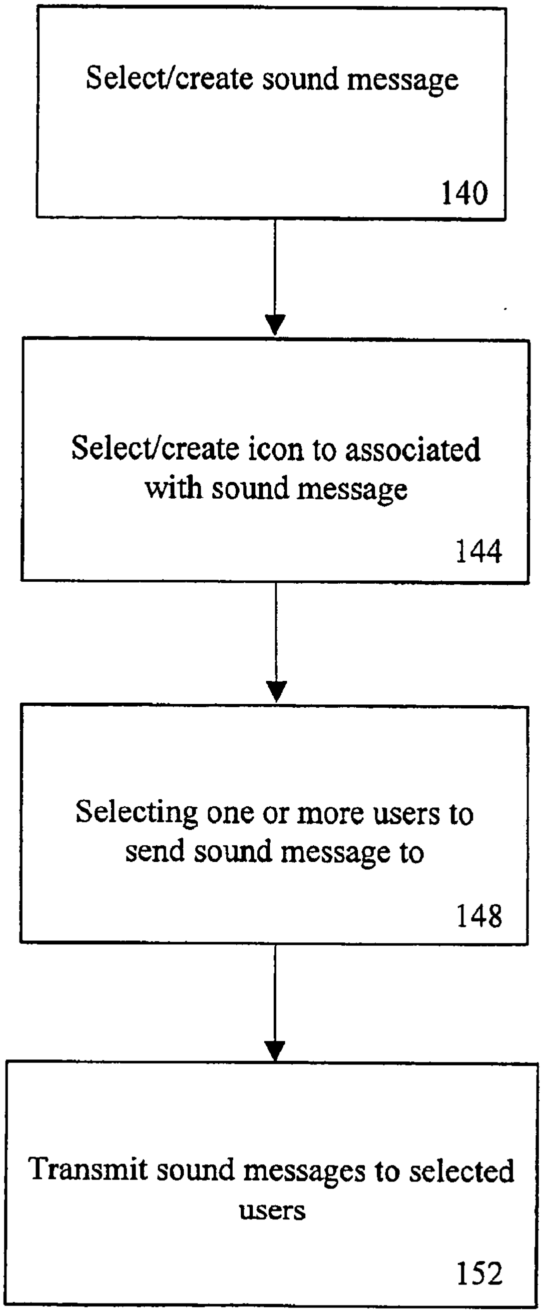

Select/create sound message
140
Select/create icon to associated with sound message
144
Selecting one or more users to send sound message to
148
Transmit sound messages to selected users
152

FIG. 3

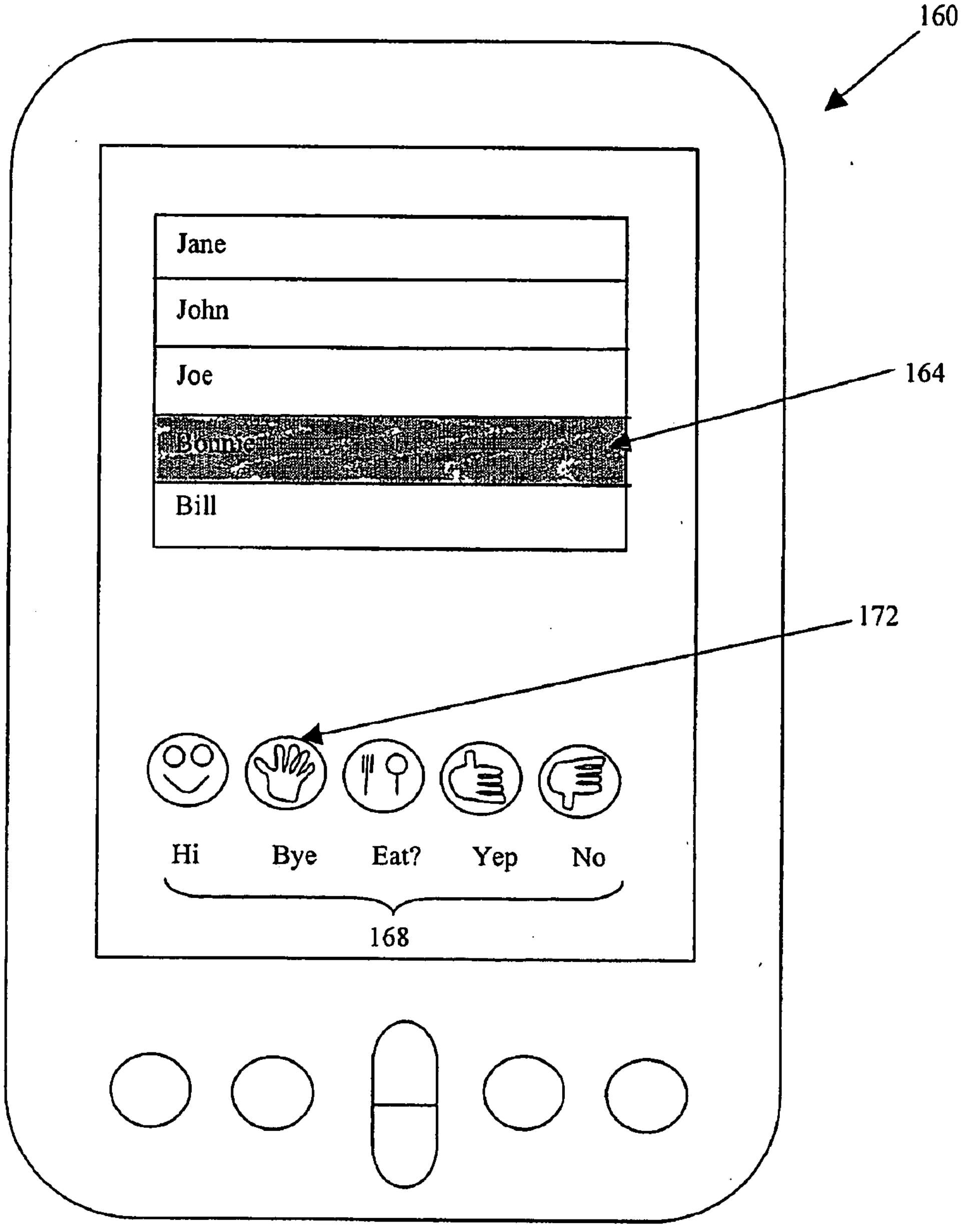


FIG. 4

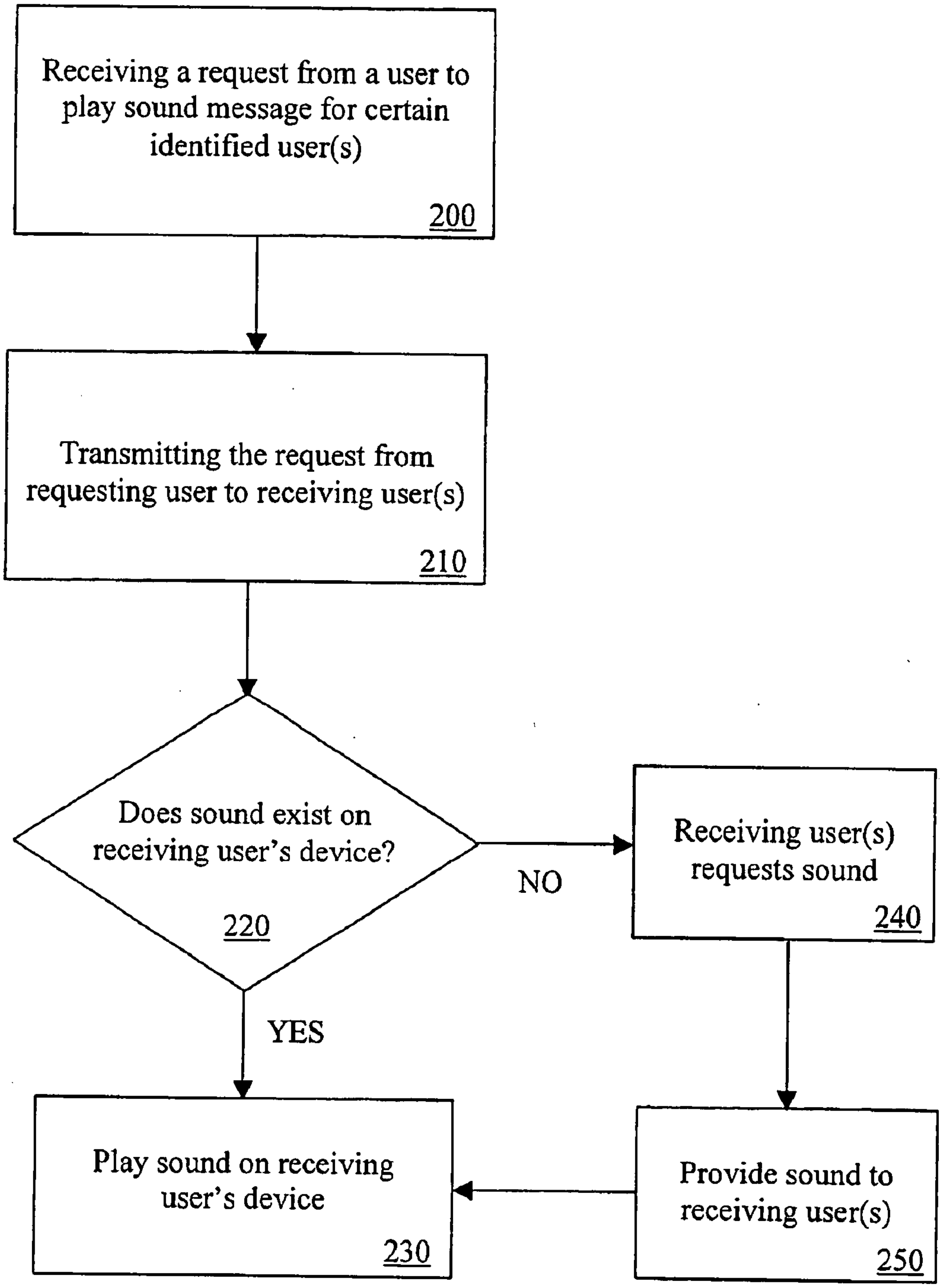

Receiving a request from a user to play sound message for certain identified user(s)
200
Transmitting the request from requesting user to receiving user(s)
210
Does sound exist on receiving user's device?
220
NO
Receiving user(s) requests sound
240
YES
Play sound on receiving user's device
230
Provide sound to receiving user(s)
250

FIG. 5

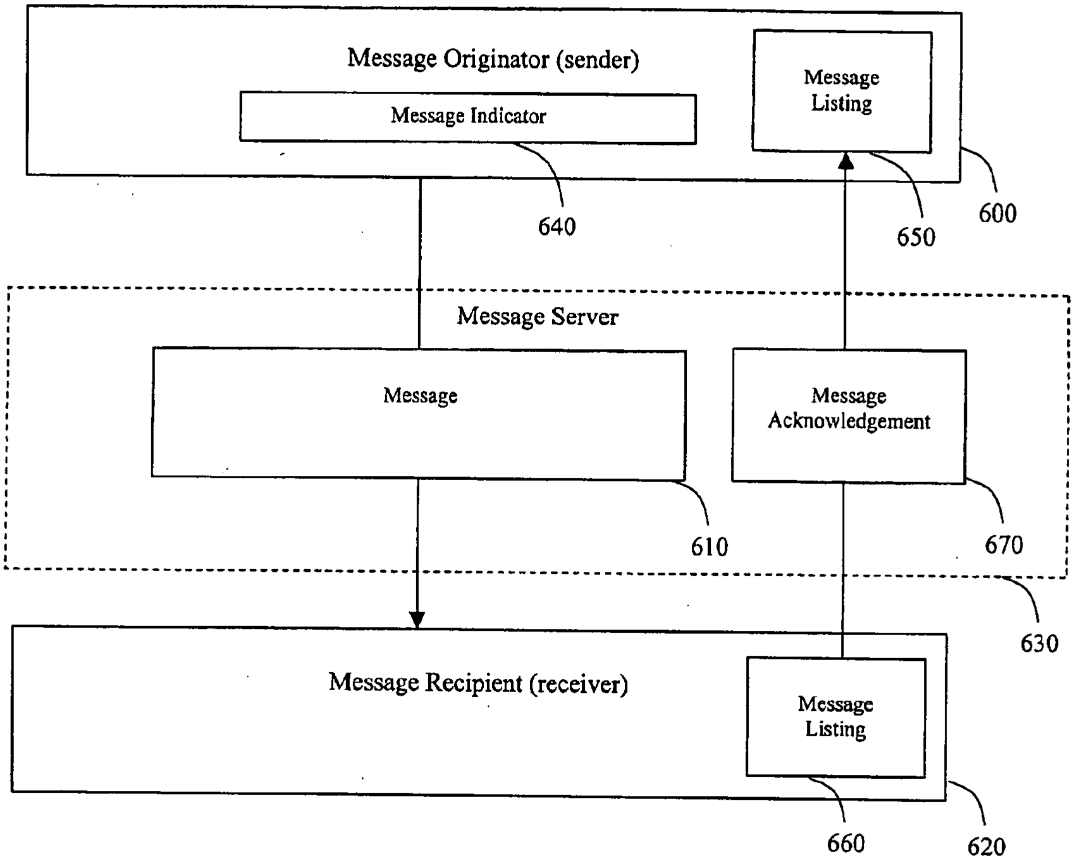

Message Originator (sender)
Message Indicator
Message Listing
640
650
600
Message Server
Message
Message Acknowledgement
610
670
630
Message Recipient (receiver)
Message Listing
660
620

FIG. 6

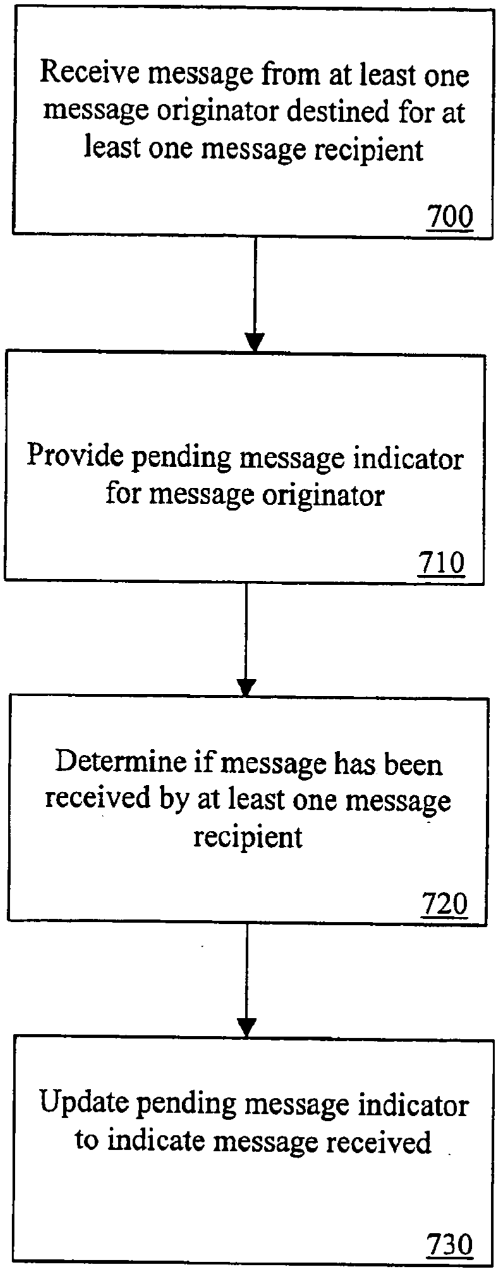

Receive message from at least one message originator destined for at least one message recipient
700
Provide pending message indicator for message originator
710
Determine if message has been received by at least one message recipient
720
Update pending message indicator to indicate message received
730

FIG. 7

SYSTEM, METHOD AND APPARATUS FOR COMMUNICATING VIA INSTANT MESSAGING

[0001] This application is a continuation in part of U.S. patent application Ser. No. 09/609,893, filed Jul. 5, 2000. This application also claims the benefit of United States provisional application No. 60/260035, filed Jan. 5, 2001 and United States provisional application No. 60/264421, filed Jan. 26, 2001, the contents of which are incorporated by reference herein.

BACKGROUND OF THE INVENTION

[0002] This invention relates to interactive communications, and more particularly, to a system, method and apparatus for communicating in a distributed network via instant messages.

[0003] One of the more beneficial aspects of the Internet, aside from the vast array of information and content sources it provides, is the varied and newfound ways people can now communicate and stay in touch with one another. Users all around the world, or even just around the corner, may now communicate in a relatively low cost and efficient manner via a myriad of Internet facilities including electronic mail, chat rooms, message boards, text based instant messaging and video tele-conferencing.

[0004] These methods of communication offer distinct advantages over standard communicative methods such as paper based mail and conventional telephone calls. For example, facilities like electronic mail are typically considerable faster and cheaper than these conventional methods of communication. Rapidly escalating in popularity is text based instant messaging which offers more instantaneous gratification with respect to interactive communications between two or more users.

[0005] However, one main problem with presently available forms of text based instant messaging and facilities like electronic mail is that both text based instant messaging and electronic mail are still both somewhat impersonal, especially compared with something like conventional telephone conversations where vocal intonation, tone and feedback provide a much needed flavor of humanity and personality to the communications. Text based instant messaging and electronic mail also typically require the users to have access to input devices such as keyboards to facilitate the creation and transmission of messages to one user from another. The quality of such communications thus depends heavily on each user's typing speed, accuracy and network connection quality of service. Furthermore, users without access to input devices such as keyboards may find it very difficult to conduct meaningful conversations without have to endure tedious keystroke input procedures.

[0006] Accordingly, it would be desirable to have a way to communicate with other users in still an efficient and quick manner but with a more personal touch than provided by other modes of electronic based communications. It would be further desirable to be able to communicate with other users on multiple devices and be able to keep track of the users on these multiple devices so that communications are not lost in the network. It would also be further desirable to be able to have users on multiple devices receive messages at their currently active client device.

SUMMARY OF THE INVENTION

[0007] The present invention is a system, method and apparatus for facilitating communications among a number of distributed users who can send and receive short sound earcons or sound instant messages which are associated with specific conversational messages. The earcons are typically melodies made up of short strings of notes. Users conversing with one another via the earcons are responsible for learning the meaning of each earcon in order to effectively communicate via the earcons. Visual aids may be provided to aid users in learning the meaning of the earcons.

[0008] In one embodiment of the present invention, the earcons are represented via visual icons on their respective communicative devices, such as their personal digital assistant devices, personal computers and/or wireless telephones. One embodiment of the present invention is a system for facilitating communication among a plurality of distributed users. The system includes a plurality of distributed communicative devices, a plurality of sound instant messages for playing on each of the distributed communicative devices and a central server which receives a request from one or more of the plurality of distributed communicative devices, transmits the request to one or more of the plurality of distributed communicative devices identified in the request wherein the one or more of the plurality of distributed communicative devices identified in the request will play the one or more of the plurality of sound instant messages also identified in the request.

[0009] The present invention is also an apparatus for facilitating distributed communications between a plurality of remote users which includes a display screen, at least one icon displayed on the display screen, the at least one visual icon associated with an earcon made up of a series of notes associated with a communicative message, and a transmitter for transmitting the earcon from the first user to at least one other user.

[0010] The present invention also is a method for communicating via sound instant messages which includes receiving one or more sound instant messages, caching the plurality of sound instant messages, receiving a request to play at least one of the cached sound instant messages and playing the at least one of the received sound instant messages from the plurality of cached sound instant messages.

[0011] The present invention further includes a method of establishing sound based communications among a plurality of distributed users in a communicative network which includes determining which of the plurality of distributed users are currently on the network, receiving a request from at least one user on the network, wherein the request identifies one or more users in the network and at least one sound instant message designated for the one or more identified users and transmitting the one or more sound instant messages to the one or more identified users in the network.

[0012] In the present invention, personal sound identifiers may accompany a sound message or earcon such that the receiving user will be alerted to the identity of the user who sent them the sound message or earcon. The earcons are typically short snippets of song riffs or some otherwise random selection of notes or sounds which are used to uniquely identify each user to one another.

[0013] The present invention is also a method for receiving a message from a message sender designated for at least one message recipient and providing status indicators as to the status of the message. In one embodiment, the method includes the steps of determining when the message is received by the at least one message recipient, wherein a determination that the message is received is confirmed by a message acknowledgement and providing a status indicator update for the message sender, the status indicator update comprising a visual representation of the message having a first appearance when the message is pending and a second appearance when the message is received by the at least one message recipient.

[0014] The message status indicator may be provided as a color or a pattern change to distinguish between the pending message status and the received message status. Message listings are created at both the sending client and the receiving client so that the sending client knows which messages have been received and the receiving client knows that the message has been seen already to discourage duplication of a message at a certain client location.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] **FIG. 1** is a diagram of an exemplary system in accordance with the teachings of the present invention.

[0016] **FIG. 2** is a diagram of an illustrative communicative device in accordance with the teachings of the present invention.

[0017] **FIG. 3** is an exemplary method in accordance with the teachings of the present invention.

[0018] **FIG. 4** is another diagram of an illustrative communicative device in accordance with the teachings of the present invention.

[0019] **FIG. 5** is another exemplary method in accordance with the teachings of the present invention.

[0020] **FIG. 6** illustrates an exemplary instant message communication setup in accordance with the teachings of the present invention.

[0021] **FIG. 7** is yet another exemplary method in accordance with the teachings of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0022] Referring to **FIG. 1**, an exemplary communications system **10** is shown in accordance with the present invention wherein users in the system may communicate with one another using sound messages or "earcons" and/or personal sound identifiers. As used herein and described in more detail later herein, the terms "sound messages", "sound instant messages" or "SIMS" and "earcons" which are used interchangeably herein, mean a short series of notes and/or sounds which are associated with or representative of any number of short communicative phrases. These short communicative phrase may be any conversational message such as "Hi", "Hello", "Are you ready to go?", "Meet you in five minutes", "I'm heading home" and a virtually infinite variety of these and other phrases. For example, a short string of six notes could be constructed to mean "Are you ready to go?" while another unique short string of four notes could be constructed to mean "Hello." Typically, each user will be provided with a basic "set" of conventional or standardized earcons which have predefined meanings such that users may readily communicate with one another using these standardized earcons without having to decipher or learn the meaning of the earcons. Additionally, new earcons may be created by each user such that when using these user-created earcons, each user is responsible for the task of interpreting and learning each other user's respective earcons in order to effectively communicate via the earcons or sound messages.

[0023] As used herein and described in more detail later herein, the term "personal sound identifier" refers to one or more short or abbreviated sound snippets which a user may use to identify themselves to another user. These sound snippets will typically be short melodies made up of short strings of notes which a user will use to identify themselves to other users in the system. The personal sound identifiers may also be snippets or riffs of popular songs, themes or melodies. Both the earcons and personal sound identifiers may be selected by a user from a predetermined selection or the sound messages and personal sound identifiers may be created by user individually, as discussed in more detail later herein.

[0024] In the present invention, text Instant Messages or "TIMS" may also be used along with or instead of the SIMS described above and/or the personal sound identifiers described below.

[0025] In one embodiment, the earcons and personal sound identifiers are used on a selective basis, whereby a user may or may not provide their personal sound identifier with each earcon sent by that user to other user(s). In another embodiment, every earcon is accompanied the user's personal sound identifier. For example, if a user's earcon is a three note melody and that user wishes to send another user an earcon which means "Are you ready to go?", the other user will hear the three note melody followed by the earcon which means "Are you ready to go?" In this manner, users can readily identify the source of the earcon which is especially valuable when multiple users are sending each other earcons during a single communicative session. Certain system rules may also be implemented regarding the playing of the personal sound identifiers. For example, if a user has received a series of earcons from a single other user, the sending user's earcon will not be played every time since it can be assumed that the receiving user is already aware of the sending user's identity. Other rules may be implemented, for example, if a user has not received any earcons for a specified period of time, such as 15 minutes, any earcons received will automatically be preceded by the sending user's personal sound identifier.

[0026] As shown in **FIG. 1**, the system **10** includes one or more communicative devices, such as personal digital assistant (PDA) devices **20**, **30**, wireless telephone **40** and personal computer **50**. In the present invention, the devices, such as personal digital assistant (PDA) devices **20**, **30**, wireless telephone **40** and personal computer **50** are in communication with one another and with a central server **60** via a plurality of communication transmissions **70**. In one embodiment, each device is associated with an individual user or client but in other embodiments, a single user or client may be associated with two or more devices in the system.

[0027] Each device may be in communication with one another and central server **60** through a wireless and/or a wired connection such as via dedicated data lines, optical fiber, coaxial lines, a wireless network such as cellular, microwave, satellite networks and/or a public switched phone network, such as those provided by a local or regional telephone operating company. In a wireless configuration, the devices may communicate using a variety of protocols including Transmission Control Protocol/Internet Protocol (TCP/IP) and User Datagram Protocol/Internet Protocol (UDP/IP). Both the TCP/IP and/or the UDP/IP may use a protocol such as a Cellular Digital Packet Data (CDPD) or other similar protocol as an underlying data transport mechanism in such a configuration. In the present invention, one to one messaging as well as multicast messaging from one user to a group of two or more users may be implemented easily via a UDP-based protocol.

[0028] In an exemplary embodiment, the devices preferably include some type of central processor (CPU) which is coupled to one or more of the following including some Random Access Memory (RAM), Read Only Memory (ROM), an operating system (OS), a network interface, a sound playback facility and a data storage facility. In one embodiment of the present invention, a conventional personal computer or computer workstation with sufficient memory and processing capability may be used as central server **60**. In one embodiment, central server **60** operates as a communication gateway, both receiving and transmitting sound communications sent to and from users in the system.

[0029] While the above embodiment describes a single computer acting as a central server, those skilled in the art will realize that the functionality can be distributed over a plurality of computers. In one embodiment, central controller **70** is configured in a distributed architecture, with two or more servers are in communication with one another over the network.

[0030] Referring to **FIG. 2**, an exemplary device for creating, storing, transmitting and receiving sound messages and/or personal sound identifiers is shown. As shown in **FIG. 2**, the device is a type of Personal Digital Assistant (PDA) **100**. It is known that PDAs come in a variety of makes, styles, and configurations and only one out of the many makes, styles and configurations is shown. In one embodiment of the present invention, PDA **100** includes a includes a low profile box shaped case or housing **110** having a front face **114** extending from a top end **118** to a bottom end **122**. Mounted or disposed within front face **114** is a display screen **126**. Positioned proximate bottom end **122** are control buttons **132**. Display screen **126** may be activated and responsive to a stylus, control pen, a finger, or other similar facility, not shown. Disposed within housing **110** is a processor coupled with memory such as RAM, a storage facility and a power source, such as rechargeable batteries for powering the system. The microprocessor interacts with an operating system that runs selective software depending on the intended use of PDA **12**. As used in accordance with the teachings herein, memory is loaded with software code for selecting/generating, storing and communicating via sound messages and/or personal sound identifiers with one or more other users in the system.

[0031] Referring again to **FIG. 2**, in one embodiment, the display screen **126** includes a screen portion **130** which displays the name, screen identification or other identifying indicia one or more other users on the network. In one embodiment, a user may be able to maintain a list of users on their device and when such as user becomes active on the network, the display will provide some indication to the user, such as by highlighting the name in some manner, to indicate that the user is available on the system. For example, an icon may appear proximate to the name of a user who is available or present on the system.

[0032] As used herein, the term "available" may include both when a user is currently "active", such as when they are presently using their communicative device or the term "available" may include when a user is "idle", such as when the user is logged on but is not currently using their respective communicative device. In certain embodiments, a different icon may be used to distinguish between when a user is in an "active" or in an "idle" state. In the present invention, clients or users via their respective communicative devices such as PDAs, laptops, PCs, etc. may update a centralized server with their presence information via a lightweight UDP-based protocol. Typically, the server will fan a client's presence information out to other users or clients that have indicated an interest and have permission to see it. Thus in a case where one user may be "logged on" on two or more devices, the sound message request will be transmitted to the user on the device which is deemed to be currently in an "active" state. In the present system, users may be alerted as to the state change of other users in the system, such as when a certain user becomes "active" or changes from "active" to "idle." Such alerts may be provided via sound-based alerts which will indicate the state changes to the users. Such alerts may be followed, for example, by the user's personal sound identifier which identifies the user who has changed their respective "state."

[0033] As shown in **FIG. 2**, the display screen **126** includes one or more visual indicia or icons **134** which are associated with one or more sound messages, sound instant messages or earcons. For example, five different representative sound icons **134** are shown, each icon associated with a distinct sound message or earcon such as "Hi", "Bye", "Eat", "Yep" and "No". To facilitate communication via the earcons, each icon may include a textual or visual label to assist the user in remembering which icon is associate with which earcon. For example, referring to the icons **134**, the "Eat" icon may includes a picture which hints as to the meaning of the earcon, such as a fork and spoon as illustrated and may also include a textual label such as "Eat?" As discussed in more detail later herein, each sound message may be user created, such as the user employing a sound creation/editing utility which the user may use to compose the earcon or the user may select from system provided earcons from which a user may make selections. Similarly, icons **134** which are associated with the earcons may be user created such as via specialized software for designing and editing bitmaps of icons and/or the icons may be provided by the system from which a user may select.

[0034] Referring again to **FIG. 2**, the display screen **126** may further include a visual log for recording and displaying the different sound message or earcons which a user may have received. Such a visual log may aid a user in learning the meaning of earcons for which the user is unfamiliar with.

[0035] Referring now to **FIGS. 3 and 4**, an exemplary method and device is shown for creating and transmitting sound messages and/or personal sound identifiers between users in the system. As shown in **FIG. 3**, the user creates a sound message, step **136**. A sound message may be created by simply selected a sound message from a selection of pre-recorded sound messages or sound message may be newly created by a user, such as by employing a sound editor utility to construct a sound message. Once a sound message is created, he sound message is saved, step **140**. Saving may be done locally on a user's personal communicative device by simply saving the sound message with, for example, a sound editor utility as a sound file on the device's storage facility. The user may then select or create an icon to be associated with the sound message, step **144**. The icon may be selected from a selection of already existing icons or may be specially created by the user via a graphics utility or facility. In other embodiments, an icon may be assigned to the sound message automatically. Once an icon is selected/created and is now associated with a specific sound message, the user may send the sound message to any number of users in the system. To accomplish this, the user may select one or more users to send the sound message to, step **148**. This may be accomplished, as discussed in more detail later herein, such as by selecting one or more user names from a directory of users. The user may then transmit the sound message to the selected users by selecting or activating the icon associated with the desired sound message, step **152**.

[**0036**] As discussed in more detail later herein, typically the file in which the sound message or earcon is stored is not itself transmitted to users directly. Preferably, each user already has a "copy" of the sound message stored or cached locally such that only a request or command to play the sound message is transmitted by the user. However, in cases where a user just created a new sound message, the sound message would first need to be distributed to the other users in the system. Preferably this is accomplished on "as-needed" basis whereby the new sound message is transferred "on-the-fly" to users who does not yet have a stored or cached version of the new sound message. For example, the user who has created the new sound message will simply send the sound message like any other sound message at which point the receiving user who does not yet have the sound message will request transfer of the new sound message.

[**0037**] In other embodiments, the proliferation and distribution of sound messages or earcons may be accomplished by having specialized software automatically distribute a new sound message to the other users when the software detects that new message has been created. In another embodiment, a central repository of sound messages or earcons may be administered via a central server, such as illustrated in **FIG. 1**. In this embodiment, the central server would maintain a central repository of all sound messages or earcons in the system and would periodically update user's devices with the earcons as new one were created. Similar methods may be used to delete sound messages or earcons which are obsolete or unwanted.

[**0038**] In the present invention, as new sound messages or earcons are created, each sound message is assigned a unique identifier, which can be a numerical identification (ID), alphabetical ID, a combination thereof or other unique identifier which is unique to that particular sound message. In this manner, sound messages or earcons are identified within the system between users via these unique identifiers.

[**0039**] In one embodiment of the present invention, the files containing the sound messages or earcons are stored locally on each user's local device such as their PDA. Sound messages may be stored as sound files in any one or other file formats such as in a MIDI file format, a .MP3 file s format, a .WAV file format, a .RAM file format, .AAC file format and a .AU file format.

[**0040**] Referring now to **FIG. 4**, an exemplary device **160** for implementing the steps as discussed above and shown in **FIG. 3** is shown. In this embodiment, a user may send one or more other users a sound message or earcon as follows. The user employing the device **160** makes a selection from a screen portion **164** which lists some identifying indicia, such as the names of system users, shown herein "Elena, Alan, Dipti, Bonnie and Maya." In an exemplary embodiment, one user say for example, "Elena", selects "Bonnie", by selecting via a stylus, not shown, the name "Bonnie" which is then subsequently highlighted. The user then taps or selects the appropriate icon from the selection of icons **168** which is associated with the sound message or earcon the user wishes to send to "Bonnie." For example, if the user wishes to send the sound message "BYE" to "Bonnie" the user will simply select the icon "BYE"**172** which will transmit the associated earcon to "Bonnie", or more specifically a command or request will be transmitted to "Bonnie" to play the earcons associated with icon **172**. "Bonnie's" respective device will then undertake playing the sound message, such as via a sound playback facility which may include a sound processor and a speaker component. In one embodiment, only the "BYE" earcon is played on "Bonnie's" device and in other embodiments, the "BYE" earcon is accompanied by "Elena's" personal sound identifier. Thus, if "Bonnie" did not already know that the earcon originated from "Elena", "Elena" personal sound identifier should provide "Bonnie" with this information. Typically, the personal sound identifier will be played before playing the earcon but the personal sound identifier may also be played after the earcon is played. In the present invention, it is contemplated that a user may send another user a series of sound message by multiply selecting two or more earcons to send to the user. In this manner, a user may actually construct phrases or sentences with a variety of independent earcons strung together. A user may also send the same earcon to multiple users simultaneously.

[**0041**] Referring to **FIG. 5**, an exemplary method for facilitating communications in accordance with the present invention is shown. In this embodiment, a command or request is received from a user to send one or more users a sound message(s) or earcon(s), step **200**. In its most basic form, a user request identifies the user or users to which the sound message is intended for, and a unique identifier or ID of the sound message to be played. As discussed above, the request may be simply the user selecting one or more names on the user's display screen and activating the icon associated with the sound messages the user wishes to send. Alternatively, the request may also include the requesting user's personal sound identifier as discussed earlier herein. The request will be transmitted to the receiving user's device, step **210**. Once the request is received, it is determined if the sound message exists on the receiving user's device, step **220**.

[**0042**] As discussed earlier herein, each user's device in the system will preferably have a locally cached or stored selection of sound messages or earcons created by other users in the system such that when one user sends another user a sound message, the sound will simply be played from the selection of locally resident sound messages. Thus, a determination if a sound message exists on the receiving user's device may be accomplished by comparing the unique identifier of the sound message contained in the request with the unique identifiers of the sound messages already existing on the receiving user's device. If a sound message does not exist on a user's device, a request for the missing sound message is made, step **240**. Ideally, specialized software on the receiving user's device will automatically administer the request for a missing sound message. The missing sound message may either be requested directly from the requesting user or from a central server which may maintain a current selection of sound messages. The missing sound message is then provided to the receiving user, step **250**. The message can then be played on the receiving user's device, step **230**.

[0043] In one embodiment of the present invention, the sound message request includes the requesting user's personal sound identifier or at least some indication as to the identity of the user sending the request. Thus, the receiving user(s) device will play the personal sound identifier along with playing the sound message. In one embodiment, each user's personal sound identifier may be distributed to other users in the system similar to the manner in which sound message sound files are distributed to users in the system and stored on their local devices. The actual personal sound identifier may also be simply transmitted along with the request as discussed above. In this embodiment, a receiving user would receive the personal sound identifier along with the request to play a certain sound message. The personal sound identifier would be played along with the stored sound message.

[0044] In another embodiment of the present invention, the playing of a user's personal sound identifier may be performed automatically by each user's device. The user's device would play a user's personal sound identifier whenever a sound message is received from that specific user. In this manner, specialized software provided on the device will determine which user has sent a sound message and then play that user's respective personal sound identifier.

[0045] In one embodiment of the present invention, the sound message communications will support message authentication, and optional message encryption. In one embodiment, authentication will likely be accomplished by including an MD5(message+recipient-assigned-token) MAC with the message. A Tiny Encryption Algorithm (TEA) for the encryption layer may also be used in one exemplary embodiment. Of course other authentication and encryption algorithms may be used.

[0046] In the present invention, each unique device such as a PDA, wireless telephone or personal computer is associated with a single user. However, at times a single user may be active on two or more devices, such that a user may communicate via the sound messages with users via the two or more devices. For example, a single user may be in communication via their PDA as well as their wireless telephone at the same time. In this manner, a display screen such as the one shown in **FIGS. 1, 2** and **4** may provide some indication that the user is on multiple devices at the same time. For example, some type of visual indicator such as a representative icon may be displayed next to the user's name to show that the user is on both their PDA and wireless telephone device simultaneously. In such an embodiment, a request or command to play a sound message will be sent to the user's device on which the user is currently active.

[0047] In the present invention, a potentially unlimited variety of communication scenarios are possible using the sound messages of the present invention, such an exemplary ritualized conversations is displayed below between a number of exemplary users where the users are exchanging a series of communicative earcons with one another:

Ann: <Earcon for "Hi!">	Bonnie: <Earcon for "Lunch?">	George: <Earcon for "Ready?">
Nancy: <Earcon for "Hi!">	Dipti: <Earcon for "Sure!">	Maya: <Earcon for "In 5">.

[0048] In this manner, users can quickly contact each other and make arrangements or just let each other know they're thinking about each other without requiring undue amounts of keystrokes, actions or input on the part of the users. Personal sound identifiers or sound identification may also be used herein to identify users to one another on the system. As discussed earlier herein, personal sound identifiers are unique abbreviated sounds which associated with specific users. For example, in the above illustrative communication, user "Ann" may have a personal sound identifier which resembles a portion of the "Hawaii-Five-O" theme song, user "Bonnie" may have a random three note melody as a personal sound identifier and user "Dipti" may have a personal sound identifier which resembles a portion of the famous song "Smoke on the Water". Thus, if user "Ann" were to send user "Bonnie" an earcon, the earcon would be preceded by the short snippet from the "Hawaii Five-O" theme song followed by the earcon to signal user "Bonnie" that the earcon was from "Ann." In conversing via the earcons, users may selectively accept and reject earcons from certain users or all users as desired. For example, user "Ann" may configure her device to accept earcons from all users, specific users such as "Bonnie" and "Dipti" or alternatively, not accept any earcons from any user. Such a configuration may be provided via specialized software on the user's respective device which allows the setting of these possible configurations.

[0049] In the present invention, only those users who have indicated a willingness or provided the necessary permission to receive such sound messages will receive such sound message. In one further exemplary scenario, exemplary USER X, USER Y and USER Z would allow each others sound messages to be propagated to one another such that USER X, USER Y and USER Z each would have a complete set of locally stored sound messages selected/created by the other users. For example, USER X would have locally saved versions of all the sound messages selected/created by USER Y and USER Z and so on.

[0050] In the present invention, a user may be logged on from as many different clients as desired, e.g. a home PC, a work PC, a laptop, and a Palm or other variations/combinations of device usage may be employed simultaneously. In contrast, other prior art Instant Messengers (IMs) may automatically log a user out as soon as the user logs in from another location (device). In the present invention, all the places where a client is logged in are tracked along with the "active" or "idle" status of the user at each respective location (device). As used herein, active means the user has used an input device, such as a mouse or keyboard on the PC, or pen taps on the Palm, within a predetermined amount of time, such as the last five minutes. As used herein, the term "idle" means the user has not used an input device for a predetermined amount of time, such as a few minutes or longer, depending on the preferences of the user and/or system administrator.

[0051] In the present invention, if a user is logged on one device and then becomes active or logs in from another device, the system automatically notices, and switches the "active device" to the new device. For example, if a user is at their desktop and then subsequently activates a personal digital assistant device, as soon as the personal digital assistant device is activated, the server notices the client's new location by having the personal digital assistant provide a signal to the server that the user is now active on that device.

[0052] In the network, a selected user's "buddies" can see where the user is through their respective system interfaces. So as soon as the selected user moves to a new active client, all of the user's buddies' interfaces update to show that now the user is now on the personal digital assistant device, whereas before the user was on their work PC.

[0053] In this embodiment, if any of the user's buddies sends the user a message, that message will automatically go to the user on the user's active client, whichever one that is. The user's buddies don't have to worry about where the user is, i.e. the user's exact active location since the message communicating process operates in a transparent fashion to the message originator or sender. The sender of the message, i.e. one or more of the buddies, can simply proceed with creating and sending their message(s) in the fashion described herein without regard to the user's active location since the server will forward the message(s) to the user's active client device, as discussed in more detail later herein.

[0054] In certain situations, where the user is "idle" on multiple client devices, a message sent to the user will be handled by providing the message to all the idle clients to ensure that the message will get to the user. In another situation where a user may be currently active on a client but then some time thereafter ceases to be active such as in a situation where the user may be at work and then leave to go home, a message may be sent to that user during that transition period, i.e. the period between when they were last active on their work device and the time they become active, on say their home device. In such a situation, the user will typically not see the message since the message was sent to the client, i.e. work device, which was perceived to be currently active. The sender of the message also may not realize that the user didn't see the message, because the user "looked active" when they sent it. The present invention resolves the preceding situation as follows: If a user receives a message at one client where they're currently active and if the user doesn't use any input devices on that client after the message arrives and then they become active on a different client, the message will be resent to the new client. If the user later becomes active on that same client, the message is not resent, since the message is already sitting there. This handles the case of a user walking out just before a message arrives and then becoming active or logging in from another client.

[0055] In the present invention, users may track the activity status of other users or buddies in the network in a number of manners. In one embodiment, when the user is in a text conversation with someone else, a window footer tells them which of three states the other party or parties are in. For example, three exemplary activity states are "X is not focused in this window", "X is focused in this window" and "X is typing in this window" where X is the party or parties. These states may appear as soon as the other party or parties moves their cursor out of the text window shared with the user, as soon as the party or parties move their cursor into that window, or as soon as the party or parties start typing, respectively. For example, if they are on the Palm, "into" or "out of" a window means they are viewing the user's IM screen or not viewing it. Users may also put an IM conversation "on hold" on the Palm so the user can go back to it, even if they go out of the window which can help users coordinate their conversations.

[0056] In one exemplary embodiment, User Datagram Protocol (UDP) is used as the messaging protocol. However, other protocols may be used to facilitate messaging between clients, such as any other Internet Protocol (IP) compatible protocol. Typically, UDP may be classified as an unreliable but lightweight message protocol. That is, messages are sent but there is no open connection between the parties, so it's possible that messages can be dropped. UDP or other similar message protocol may be used which is more suitable to wireless connections in embodiments employing wireless devices since these are likely to have communications that are to be frequently broken and re-established. In the present invention, certain mechanisms are implemented to increase the likelihood that messages will arrive without paying the cost of maintaining and re-establishing an ongoing connection, which typically consumes valuable CPU and bandwidth and affects performance.

[0057] Referring to **FIG. 6**, an exemplary messaging configuration is shown. In this embodiment, a message originator or sender **600** sends a message **610** to at least one message recipient or receiver **620**. Message **610** is send via a message server **630** which receives message **610** from message sender **600** and provides message **610** to message recipient **620**. Once message **610** is received by message recipient **610**, message recipient **620** provides a message acknowledgement or ACK **670** back to message sender **600**. A message listing **650** may be updated by message sender **650** once the ACK **670** is received from message recipient **620** while a message listing **660** may be updated by message recipient **620** once message **620** is received. Updating message listing **660** by message recipient **620** prevents messages being duplicated, such as in the case of where message ACK **670** is not received by message sender **600** and consequently, message sender **600** re-sends another copy of message **610** to message recipient **620**. In such an example, the re-sent message will be compared with the message listing by message recipient **620** and will be discarded if the re-sent message has already been tagged as being seen, as discussed in more detail later herein.

[0058] In a typical messaging exchange, there are at least four hops between which a message can be dropped. It is possible for someone to receive a message but for the sender not to know this because the acknowledgement may be dropped on the way back to the sender. And of course it is possible for a message not to arrive at the recipient. If either party has a poor connection to the server, it's not uncommon for the message or the ACK to get dropped.

[0059] In conversation, it's critical for both parties to know what they mutually know. It can cause a lot of confusion if one party thinks they've said something when in fact the other person has not seen the message. In the present invention, the user interface via a message status indicator helps the users know what was seen by both parties and what might not have been. When the user sends a message, that message appears in the text area in a "pending" style, to indicate it has not yet been received, and then changes to a "final" or "received" state to indicate that it has been received. In one embodiment, the message appears in gray type when the message is "pending", and then it switches to a certain color, such as blue, when the client gets an ACK. In a situation where an ACK never arrives, the text stays gray. On a personal digital assistant device, the message may appear inside curly brackets and the brackets are removed when the ACK arrives. Other variations of the pending and received status indicators may be implemented, for example, in a pending status, the message status indicator may appear in a certain first pattern or color, while in a received status, the message status indicator may appear in a second pattern or color which is distinguishable from the first pattern or color. In another embodiment, the message may not be visible at all when the message is pending.

[0060] In this embodiment, it is not possible for someone to believe that they said something when the other person didn't see it, but it is possible for a recipient to see a message while the sender thinks they'd didn't. This can cause just as much of a problem in a conversation. In such a situation, the present invention provides certain safeguards to prevent this problem. For example, when a client sends a message, it waits to see if it gets an ACK for a predetermined number of seconds, such as, for example, anywhere from one to ten seconds. If it does not, it resends the message. It does this as many as X times, stopping as soon as it gets an ACK. X may be any number, such as in the range of one to fifty, depending on the requirements desired. On the other end, if a message arrives that the client has already received, it sends back an ACK but it does not re-display the message. In this way, each client properly indicates whether the message got through to the other end, but no message will appear multiple times on the recipient's screen. It is possible, though, for the messages to appear in a different order on both sides. If, for example, the sender sends two messages and the first one is dropped along the way, the first message will be resent after an X number of seconds, and it will be displayed on the recipient's screen after the first.

[0061] With this approach, if someone sees that a message they sent is pending, i.e. in gray and it stays pending, i.e. gray, they can be pretty confident that the other person did not get the message. This can happen because of a number of situations, such as if the sender's connection is poor or because the recipient's is poor. To help distinguish these cases, cues are provided to the user about their own level of connectivity and to let them know if the other person has gone offline, as described in more detail later herein.

[0062] A more detailed explanation of the messaging process of the present invention now follows with associated pseudo code to represent the methodologies involved. In the present exemplary embodiment, messages are currently sent via UDP packets, however, any type of packetized transport would be appropriate. In this packetized environment, acknowledgements or "ACK" are used to verify the receipt of the messages. Additionally, each message in the protocol is assigned a sequence number. Each client assigns monotonically increasing sequence numbers to messages it initiates, with each client keeping its own sequence. For example, say an exemplary Client A wants to send a message like a Text Instant Message "TIM" to an exemplary Client B. In this embodiment, this exchange may be represented as follows:

```
Client A sends [TIM "hi" (seq #100)] -> Server -> Client B
Client B sends [ACK "for #100" (seq #34)] -> Server -> Client A
```

[0063] It is conceivable that either the TIM or the ACK could get lost in transit, e.g. dropped due to interference in the network connection. Thus, when Client A sends the TIM, it also copies the message to a list of messages awaiting ACKs. Client A then waits for the ACK from Client B for that message as may be represented by the following pseudo-code:

```
sendMessage(message, clientB) {
send message to clientB;
copy message to list_of_messages_waiting_for_acks;
}
```

[0064] When Client B receives the message it sends an ACK back to Client A. In this embodiment, the ACK data contains the sequence number of the original TIM message so that Client A knows which of its messages have been ACKed. Client B also adds the incoming messages to a list of messages already seen, as explained in more detail later herein.

[0065] When Client A receives the ACK message, Client A updates its local display, e.g. the status indicator text goes from gray to blue indicating that Client B received the message, and the message is removed from the list of messages awaiting ACKS. Thus, the receipt of the ACK triggers the sending client to update its message list and consequently the message status indicator to a "received" status, as may be represented by the following pseudo-code:

```
receiveACK(ACK) {
 foreach message in list_of_messages_waiting_for_acks {
```

-continued

```
  if (the ACK matches the message in the list) then {
    remove this message from the list_of_messages_waiting_for_acks;
    update screen;
   }
  }
}
```

[0066] If Client A does not receive an ACK within a predetermined amount of time, say for example, anywhere from 1 to 30 seconds, Client A will resend the original message. This means that the Client A is periodically walking through the list of messages waiting for ACKS, as may be represented by the following pseudo-code.

```
checkWaitingMessages() {
  for each message in list_of_messages_waiting_for_acks {
    if (message was sent > 3 seconds ago) then {
      re-send message;
    }
  }
}
```

[0067] During operation, it is possible that an ACK might get lost. For example, if Client A sends a message to Client B, and Client B responds with an ACK that is then lost on the way back to Client A, Client A is going to resend its original TIM message again in 3 seconds. Since Client B has already received and displayed the TIM, it is preferable to make sure Client B doesn't display it again. To handle this, each Client keeps a list of the sequence numbers and senders of the last set of messages that it's received. This message listing may be compiled on a threshold limit basis whereby an X number of messages are kept in the message listing, where X is a predetermined number of message, such as anywhere from 1 to 1000. Additionally, the message listing may be kept on a time threshold basis where the messages are kept in the message listing based on a predetermined time limit, such as all message in the last minute, last five minutes, etc.

[0068] To further describe the operation of the present invention, when a client receives a message, it responds with an ACK (as it has to do each time) and it checks the sequence number and sender ID to see if it's seen this message before. If the client hasn't seen it before, it processes it (displays it, plays it, whatever). If it has seen it before, the message is simply discarded. In either case, it has already sent an ACK back to Client A so Client A can stop re-sending it, as may be represented by the following pseudo-code.

```
receiveMessage(message)
{
  respond with ACK for this message;
  if (message in list_of_messages_already_seen) then {
    discard message;
  } else {
    process message; // update display, whatever
```

-continued

```
    add message to list_of_messages_already_seen;
  }
}
```

[0069] The present invention also includes a method for resending messages to the next active client, so that if for example, a user switches devices, or logs on somewhere else, e.g. at another client device, the user will get messages the user might have otherwise missed.

[0070] In the present invention, all messages go through a server, as described and shown earlier herein. Typically, a message comes from a client addressed to a specific user. Since a user can be logged on from multiple locations, e.g. multiple client devices, the server must decide which of that user's clients is the best one to send the message to. To do this, the server uses the concept of the "last active client" as well as looking at whether all the user's clients are idle.

[0071] In the present invention, clients periodically update the server on their current activity state or how 'active' they are. This may be described by simply how much the user has used the mouse, keyboard, stylus or other input device on that machine in a predetermined time frame, such as in the last ten seconds. As used herein, the "last active client" is the client that most recently reported activity, e.g. a keystroke, mouseclick, stylus selection, etc. In the present invention, no recent activity may mean that the client is "idle."

[0072] Generally, the server may decide to route a message as represented by the following pseudo-code:

```
serverSendMessage(message, user) {
  if (all clients of user are idle) then {
    send message to all clients of user;
  } else {
    send message to last active client of user;
  }
}
```

[0073] There is a situation where the user may not receive the message in accordance with the above delivery methodology. For example, if someone sends a message to a user immediately after the user leaves their currently active client, i.e. the user's work PC for the night, the server is going to send the message to the user's work PC since it appears that the work PC is an active client. When user gets home and either becomes active on their home machine, it would be desirable to see the messages that were sent to the user at my office since the user left. Otherwise, the message will remain unread at the work PC client until, for example, the user gets to work the next morning. So in this situation, when the user becomes active on the home client, the server resends me the messages originally sent to the office client.

[0074] When the server sends a message to a client, it copies it to the list_of_messages_sent and notes the client that it was sent to. If the message was sent to multiple clients (as it might have been if they were all idle), all of those clients are noted. It keeps this list of messages sent so that it can resend them if a client other than the one(s) it was sent to becomes active next, as may be represented by the following pseudo-code:

```
serverSendMsg(message, user) {
  if (all clients of user are idle) then {
    send message to all clients of user;
    copy message to list_of_messages_sent;
    copy all clients to list_of_clients_this_message_was_sent_to;
  } else {
    send message to last active client of user;
    copy message to list_of_messages_sent;
    copy last active client of user to list_of_clients_this_message_sent_to;
  }
}
```

[0075] In the present invention, messages are removed from the list_of_messages_sent when a client in the_list_of_clients_this_message_sent_to reports in with an activity>0. This means that there is keyboard/mouse/stylus activity on that client, which means that the user must still be there and has seen the message. If another client (of that User) reports in with activity>0 next, then that means that the User must have switched devices (or logged on from somewhere else), and we need to resend the message to that (newly active) client.

[0076] So, whenever any client reports in with activity, the server performs the following as may be represented by the following pseudo-code:

```
handleClientActivity(client)
{
  if (client activity > 0) {
    // See if there are any messages we need to send to
    // this (possibly newly active) client.
    for each message in list_of_messages_sent {
      if (message was sent to this client) then {
        remove this message from list_of_messages_sent;
      } else {
        // this message was sent to another of our clients
        // but we're the first to report activity
        send message to this client;
        remove this message from list_of_messages_sent;
      }
    }
  }
}
```

[0077] Note that when a message is re-sent to another client, the client also tries to provide a rough indication of when the original message was sent. For example, if it takes a certain user two hours to get home from work and the user subsequently becomes active on their home PC, the server will provide a message like "[The following messages were originally sent to you a few hours ago]", followed by the messages that were sent to the user's work PC right after the user left.

[0078] Referring to **FIG. 7**, one embodiment of the present invention is shown. In this embodiment, a message is received from at least one message originator destined for at least one message recipient, step **700**. A pending message indicator is provided for message originator, step **710**. It is determined if the message has been received by at least one message recipient, step **720**, as may be determined in accordance with the descriptions above. The pending message indicator is updated to indicate message received, step **730**. Updating the message indicator may be performed as described earlier herein, for example, by changing the message indicator from a first pending appearance, to a second received appearance as may be evidenced by a color or pattern change or other distinguishable appearance change.

[0079] In the present embodiment, clients typically do not have continuous connections to the server, so it is impossible to know for certain when a client is offline. However, every client provides updates to the server every X number of seconds, where X is a number, such as in the range from zero to one hundred and twenty seconds. Such updates contain information about the client's status, and in return, the server sends back status information about each of the buddies or "bubs" for the client. In this embodiment, if a client does not send any updates for one minute, the server marks that client as offline.

[0080] In one situation, if a user is in a conversation with one or more other parties and the one or more other parties go offline, within a minute, the server will mark them as offline, and send a message to the user's client. The conversation window with the one or more parties will display a message "[{PARTY} is offline]". If the one or more parties later comes back online and the user has kept a conversation window with the one or more parties open, a new message will appear saying "[{PARTY} is back online]." Even if the user has closed that window, a status window displays when the one or more parties goes offline and indicates visually and with sound when they come back online.

[0081] To help the user know if they are connected from a personal digital assistant device or any other device, a visual indicator is provided of whether the user is connected. For example, there is an icon that appears on all screens of the interface that has two states: Connected and Connecting. If, for example, a user is not connected but is are running the system, the system will continue to try to connect. Since the client is sending a message to the server every X number of seconds, any time the client does not receive its return message from the server, the "Connected" icon changes to "Connecting," to indicate that there may be a problem with the connection. If it receives the return message after the next update, the icon returns to connected. If not, it stays "Connecting."

[0082] The following is an explanation of what happens if an exemplary user is in a conversation with someone and the user loses connectivity. First, each time the user sends a message, the message will appear in the "pending" style. After X number of seconds, the user's icon will change to

Connecting rather than Connected. The user will also receive no new incoming messages. If the icon stays Connecting for a while and the user receives no confirmations of the user's messages, the user can conclude that the connection is bad. If, however, the other person or parties has lost connectivity while the user are still connected, then the pattern will be different. The user's messages will appear in the "pending" style and the user won't get incoming messages but the user's icon will show the user as Connected. After a minute, a message will appear in the conversation window saying that the other person or parties has gone offline. If the other party or parties reconnects before that minute is up, then the user would see the user's "pending message" switch to received and new incoming messages would arrive, since the other party's client would be trying to resend them.

[0083] It will be apparent to those skilled in the art that many changes and substitutions can be made to the systems and methods described herein without departing from the spirit and scope of the invention as defined by the appended claims.

We claim:

1. A method for communicating via instant messages, the method comprising:

receiving a message from a message sender designated for at least one message recipient;

determining when the message is received by the at least one message recipient, wherein a determination that the message is received is confirmed by a message acknowledgement; and

providing a status update for the message sender, the status update comprising a visual representation of the message having a first appearance when the message is pending and a second appearance when the message is received by the at least one message recipient.

2. The method of claim 1, wherein determining when the message is received by the at least one message recipient comprises:

assigning a unique sequential number to the message; and

updating a message listing, wherein the message is identified by the unique sequential number in the message listing.

3. The method of claim 1, wherein determining when the message is received by the at least one message recipient comprises:

assigning a unique sequential number to the message; and

providing an acknowledgement to the message sender when the message is received by the at least one message recipient, the acknowledgement identifying the unique sequential number of the message.

4. The method of claim 3, wherein the acknowledgement is also assigned a unique sequential number.

5. The method of claim 1, wherein the visual representation is text from the message that alternates between a first color in the first appearance and a second color in the second appearance.

6. The method of claim 5, wherein the first color indicates the message is pending receipt and the second color indicates the message is received.

7. The method of claim 1, wherein the visual appearance alternates between a first pattern and a second pattern, at least one of the first and the second pattern indicating the message is pending and the other indicating the message is received.

8. The method of claim 1, wherein determining when the message is received by the at least one message recipient comprises:

providing an acknowledgement to the message sender when the message is received by the at least one message recipient, the acknowledgement identifying the unique sequential number of the message, wherein the visual appearance of the status update before the acknowledgement is received is provided in a first configuration and in a second configuration once the acknowledgement is received.

9. The method of claim 1, further comprising:

sending the message to the at least one message recipient to a plurality devices used by the message recipient.

10. A method for establishing instant messaging communications between a message originator and one or more message receivers, the method comprising:

receiving an instant message from the message originator;

providing the instant message from the message originator to the one or more message receivers specified by the message originator; and

providing an update of a local message display of the message originator once the message originator's instant message is received by the one or more message receivers, wherein the local message display is provided in a pending configuration before the message is received by the one or more message receivers and the local display is provided in a received configuration once the message is received by the one or more message receivers, the receipt of the message being acknowledged by a message acknowledgement sent by the one or more message receivers.

11. The method of claim 10, wherein the local message display provides a display of the message text in the pending and received configurations.

12. The method of claim 10, further comprising:

determining when the instant message is received by the one or more message receivers.

13. The method of claim 12, wherein determining when the instant message is received by the one or more message receivers comprises:

establishing a message listing for the message sender and the one or more message receivers, wherein the message listing updated based on an acknowledgement of the message being received by the one or more message receivers.

14. The method of claim 13, wherein the message listing prohibits duplication of the instant message on the one or more message receivers' devices.

15. The method of claim 13, wherein each message in the message listing is identified by a unique serial number.

16. A computer readable medium having instructions stored thereon for executing the steps comprising:

creating a message listing of one or more pending messages;

providing a message indicator in a first appearance for the one more pending messages;

updating the message listing based on the receipt of message acknowledgements; and

providing the message indicator in a second appearance for the one or more pending messages as the pending messages are deemed to be received.

17. The computer readable medium of claim 16, wherein the first appearance and the second appearance correspond respectively to a first coloration and a second coloration of the message.

18. The method of claim 16, further comprising:

determining the location of one or more intended recipients of the pending messages.

19. The method of claim 18, further comprising:

removing a message from the message listing when the message is deemed received by the message acknowledgement.

20. The method of claim 16, further comprising:

compiling a message listing of all pending messages wherein the message listing is updated when a message is acknowledged as received.

* * * * *